# ПРОГРАММИРОВАНИЕ НА SCRATCH. ЧАСТЬ 2.

ПОДРОБНОЕ ПОШАГОВОЕ РУКОВОДСТВО ДЛЯ САМОСТОЯТЕЛЬНОГО ИЗУЧЕНИЯ РЕБЕНКОМ

ДЛЯ СРЕДНЕГО ШКОЛЬНОГО ВОЗРАСТА

Д.В. ГОЛИКОВ, А.Д. ГОЛИКОВ

SHOUT
SHARE
IT

- О ЧЕМ ЭТА КНИГА.
- БЛАГОДАРНОСТИ.
- ССЫЛКИ.
- КАК РАБОТАТЬ С КНИГОЙ.
- ВВЕДЕНИЕ.
- ГЛАВА 1. ФЕРМА.
- ГЛАВА 2. АТАКА ЗОМБИ.
- ГЛАВА 3. СОБИРАЕМ ЯБЛОЧКИ.
- ГЛАВА 4. СТРИТРЕЙСИНГ.
- ГЛАВА 5. КОСМИЧЕСКАЯ БИТВА.
- ГЛАВА 6. ТАНЦЕВАЛЬНЫЙ КОВРИК.
- ГЛАВА 7. ДИВЕРСАНТ.
- ГЛАВА 8. БИТВА ЗА ДЕНЬ РОЖДЕНИЯ КОТА.
- ПРИЛОЖЕНИЕ 1. ОСНОВНЫЕ ПОНЯТИЯ SCRATCH 2.
- ПРИЛОЖЕНИЕ 2. СПРАВОЧНИК ПО БЛОКАМ SCRATCH 2.0.
- ПРИЛОЖЕНИЕ 3. ОТРИЦАТЕЛЬНЫЕ ЧИСЛА.
- ПРИЛОЖЕНИЕ 4. ГРАДУСЫ.
- ПРИЛОЖЕНИЕ 5. КООРДИНАТЫ X И Y.
- ПРИЛОЖЕНИЕ 6. ДЕСЯТИЧНЫЕ ДРОБИ.
- ПРИЛОЖЕНИЕ 7. ПРОЦЕНТЫ.
- ПРИЛОЖЕНИЕ 8. ФУНКЦИИ.
- КРАТКОЕ СОДЕРЖАНИЕ СЛЕДУЮЩЕЙ КНИГИ.


**Программирование на Scratch 2.**
**Часть 2.**

Делаем сложные игры.

Подробное пошаговое руководство для самостоятельного изучения ребёнком.

Голиков Денис и Голиков Артём

Copyright Голиков Денис и Голиков Артём 2014
Al rights reserved.
Лицензионные Соглашения

Благодарю вас за загрузку этой бесплатной электронной книги. Хотя это - бесплатная книга, но она остается охраняемой авторским правом собственностью автора, и не может копироваться и распространяться в коммерческих целях. Если Вам эта книга понравилась, пожалуйста, посоветуйте друзьям загрузить свою собственную копию. Благодарю вас за вашу поддержку.

# О ЧЕМ ЭТА КНИГА.

Целью книги является пропедевтика программирования, математики, естественных наук среди школьников средних классов. Книга написана на основе опыта обучения программированию на Scratch в кружке юных программистов.

Материал рассчитан на самостоятельное изучение школьниками 5–7 классов, хорошо знающими математику.

Scratch — это новая бесплатная среда программирования, которая позволяет младшим школьникам создавать собственные истории, мультфильмы, игры, презентации и другие произведения. Scratch специально разрабатывался как новая учебная среда для обучения школьников программированию. Программы создаются из разноцветных блоков точно так же, как собираются из разноцветных кирпичиков конструкторы Лего.

Scratch разработан в 2006 г. под руководством Митчела Резника группой Lifelong Kindergarten в лаборатории Media Lab Массачусетского технологического института.

# БЛАГОДАРНОСТИ.

Спасибо создателям Scratch за волшебную возможность заниматься творчеством в новом формате и обучать детей новейшим технологиям.

Огромная благодарность Марии Конопатовой за комиксы, обложку и иконки Котёнка. Все её работы можно найти на сайте http://komixsisters.ru.

Спасибо ребятам, которые помогали тестировать главы книги.

Спасибо тем, кто не оставался равнодушным, и направлял критические замечания.

# ССЫЛКИ.

Сайт авторов, посвящённый обучению детей программированию на Scratch. http://scratch4russia.com.

Все примеры из книги опубликованы на сайте http://scratch.mit.edu/users/scratch_book/

Вопросы авторам можно задавать по почте scratch.book@ya.ru.

Страничка в Фейсбук: http://www.facebook.com/ScratchBook4u

Страничка Вконтакте: http://vk.com/scratch.book

# КАК РАБОТАТЬ С КНИГОЙ.

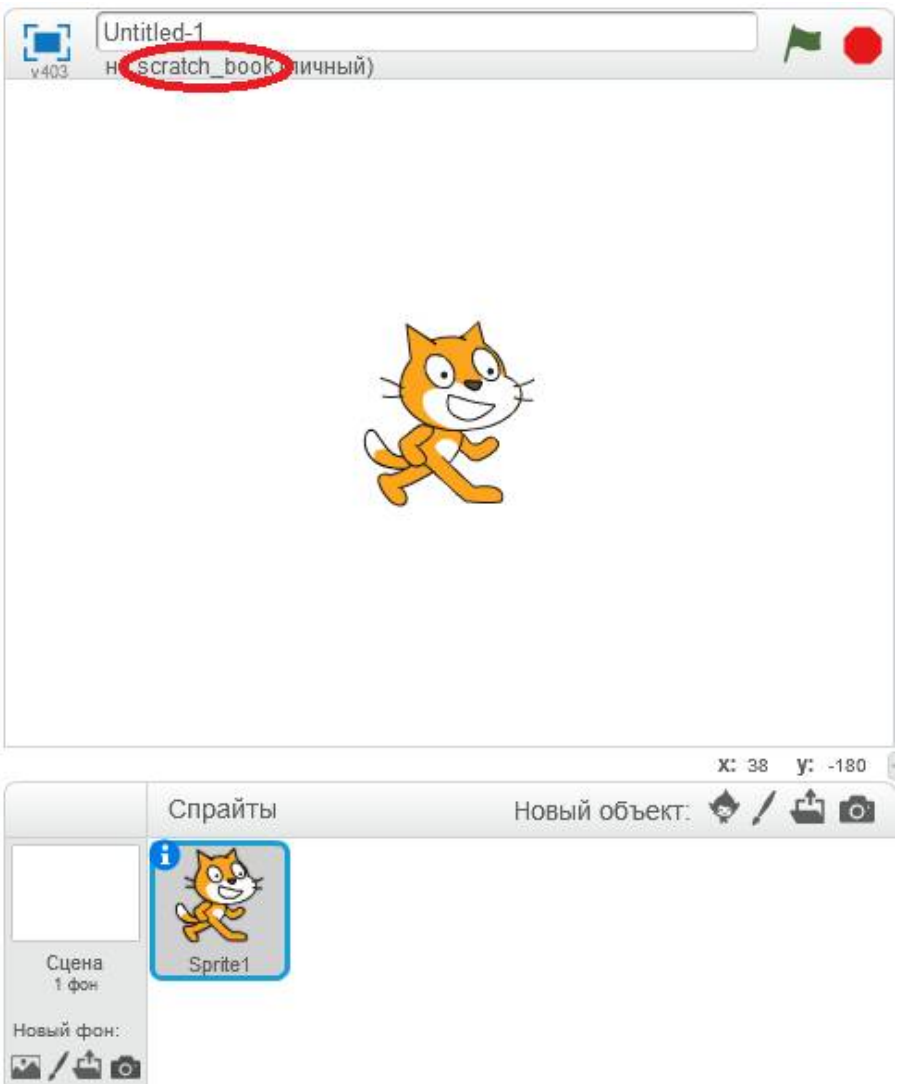
Основной принцип обучения ребёнка программированию по этой книге состоит в самостоятельном создании школьником сложной игры, состоящей из многих небольших конструкций — скриптов. Принцип работы каждого скрипта изложен очень подробно. После создания игры, ученик может некоторое время поиграть в игру, и, тут же внести изменения в игровой процесс, упростить или усложнить игру.

После каждой главы даны небольшие вопросы, которые помогут ребёнку проверить, всё ли он понял в работе игры, а также несколько заданий по её доработке. Как Вы уже можете догадаться, не смотря на то, что каждая игра в книге является законченной программой, её совершенствование может продолжаться бесконечно. Это поможет начинающему программисту научиться работать над большими и сложными приложениями, поможет развить память и аналитический склад ума.

Не стоит забывать, что создание игр потребует от вашего школьника умения рисовать на компьютере. В ранних версиях Scratch существовал только простейший редактор изображений, похожий на стандартный редактор Paint, однако, в версии Scratch 2, о которой пойдёт речь в этом учебнике, к простейшему растровому редактору, к огромной радости детишек, добавлен векторный редактор. Это многое изменило, игры стали выглядеть опрятней, проще стало изменять персонажей, в процесс обучения добавилось больше веселья. Изучению и приёмам работы в векторном редакторе в книге уделено очень много внимания, не буду скрывать, некоторым ребятам создание векторных персонажей даётся нелегко, зато, после освоения приёмов рисования, позволяет хвастаться перед друзьями, профессионально сделанными играми. Стоит заметить, что все игровые персонажи легко модифицируются, и могут быть изменены учениками без уведомления автора. На занятиях в «Клубе юных программистов», только самые ответственные дети стремятся сделать персонажей точно, как в учебнике, большая часть пытается нарисовать всё по-своему. Если программирование

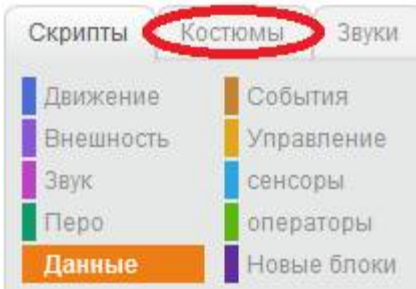
поведения игровых объектов некоторым ребятам не позволяет с лёгкостью включать фантазию и сразу же модифицировать программу по своему усмотрению, программы всё же довольно сложные, то в процессе рисования фона и персонажей фантазируют почти все. Чего только не делали они с бедным котиком!

Некоторые допущения, на которых я нигде не буду останавливаться в процессе изложения. Каждая игра создаётся из стандартного пустого проекта, в котором сидит кот. Вы должны просто запустить Scratch 2 путем запуска ярлыка на рабочем столе, или посетив интернет по адресу http://scratch.mit.edu. При запуске программы Вы всегда увидите пустой проект. Красным обведено имя пользователя.



Во многих играх котик нам пригодится, он будет главным участником интересных событий, а в некоторых нам придётся его удалить. Создавать игры, как правило, мы будем, начиная с рисования фонов и спрайтов. Как Вы уже знаете из первой части книги, спрайтами называются персонажи игры. Их можно рисовать самим, либо загружать из библиотеки спрайтов. Нажав фотоаппарат, можно загрузить в компьютер даже своё собственное изображение. Это очень весёлая функция программы, однако, в этой книге она описываться не будет.

Когда я буду говорить: «Зайдите в графический редактор» - надо будет просто кликнуть на вкладку «Костюмы».




На вкладке Костюмы расположен графический редактор, и все костюмы спрайта. Красными овальчиками я всегда буду выделять те места на картинках, на которые надо кликнуть мышкой.

этой книге, в отличие от множества других книг по программированию не будут даваться объяснения к каждому блоку. Названия всех блоков Scratch говорят сами за себя. Но, на всякий случай, в приложении будет приведён коротенький справочник с описанием каждого блока. Это поможет тем новичкам, которые не читали первую книгу, а принялись сразу за создание сложных игр.

# ВВЕДЕНИЕ.

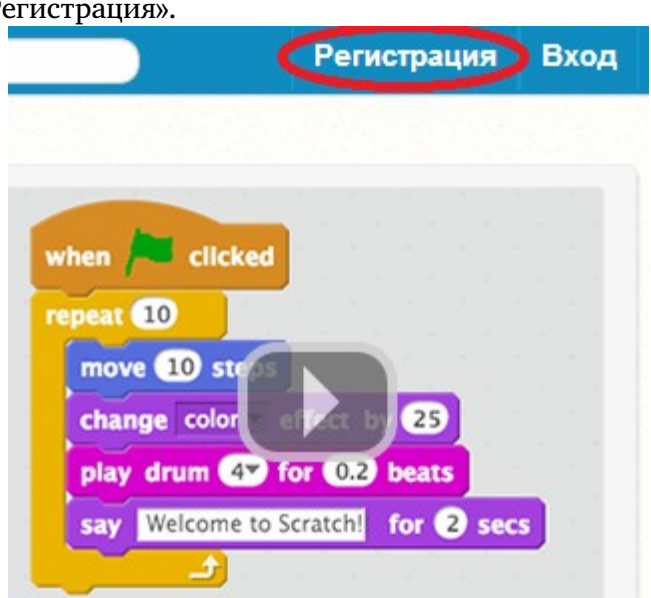
Существует два способа работы в Scratch. Можно работать в браузере, а можно установить оффлайновый редактор и делать программы не подключаясь к интернет. Если Вы планируете работать используя интернет, то прочитайте первую часть введения. Если Вы планируете работать без подключения к интернет, то можете перейти сразу ко второй части введения.

Обратите внимание, все примеры проектов в этой книге сделаны в оффлайновом редакторе.

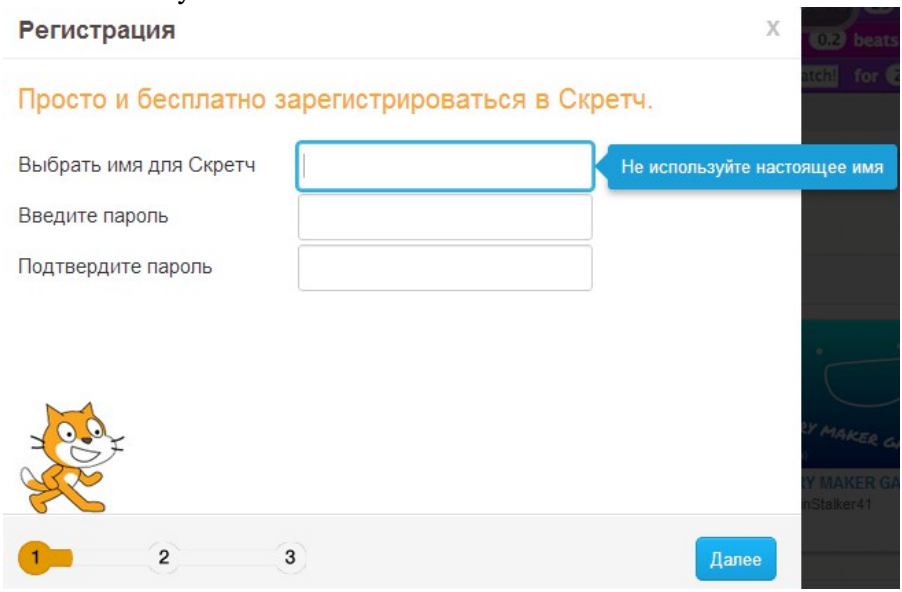
РАБОТА НА ПОРТАЛЕ SCRATCH.MIT.EDU.

Для получения возможности создавать свои проекты на Scratch, Вам необходимо зарегистрироваться на сайте scratch.mit.edu. Для регистрации Вам нужно иметь свой адрес электронной почты, попросите родителей Вам его завести. Когда у Вас появится свой адрес электронной почты, то первым делом запишите его в блокнот. Пароль тоже лучше записать, чтобы случайно не забыть.

Теперь перейдите на сайт http://scratch.mit.edu и нажмите на кнопку «Регистрация».

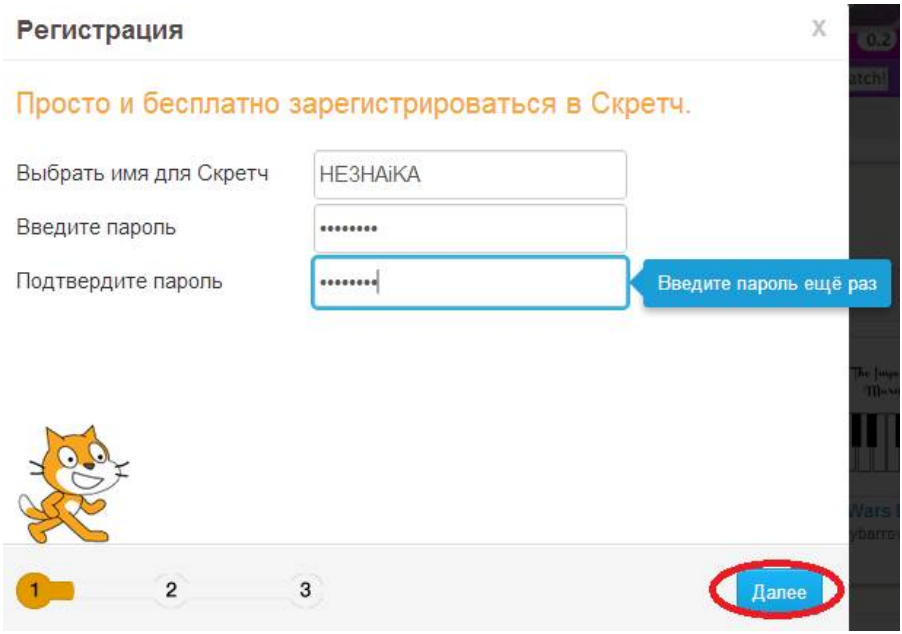


Откроется окно регистрации, в котором в верхнее окно надо ввести Ваше имя для сайта Scratch (иногда это имя называют «логин»). Разработчики советуют не вводить настоящее имя, я тоже не советую.



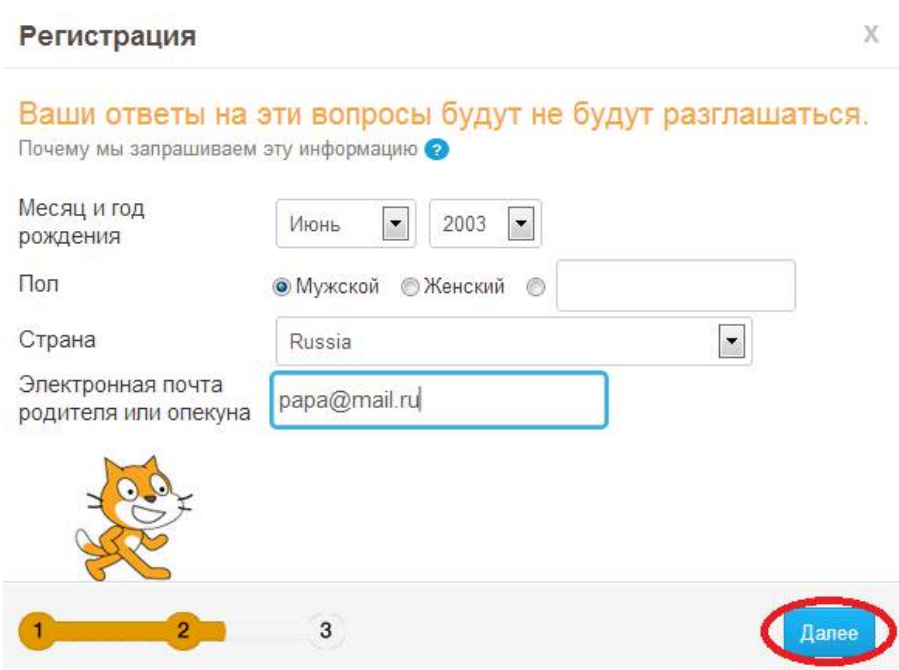

Придумайте интересный логин, например, типа такого, здесь использованы буквы латинского алфавита и цифра «3» вместо буквы «з».

Затем два раза введите пароль и нажмите кнопку «Далее». Не забудьте записать пароль в блокнот!

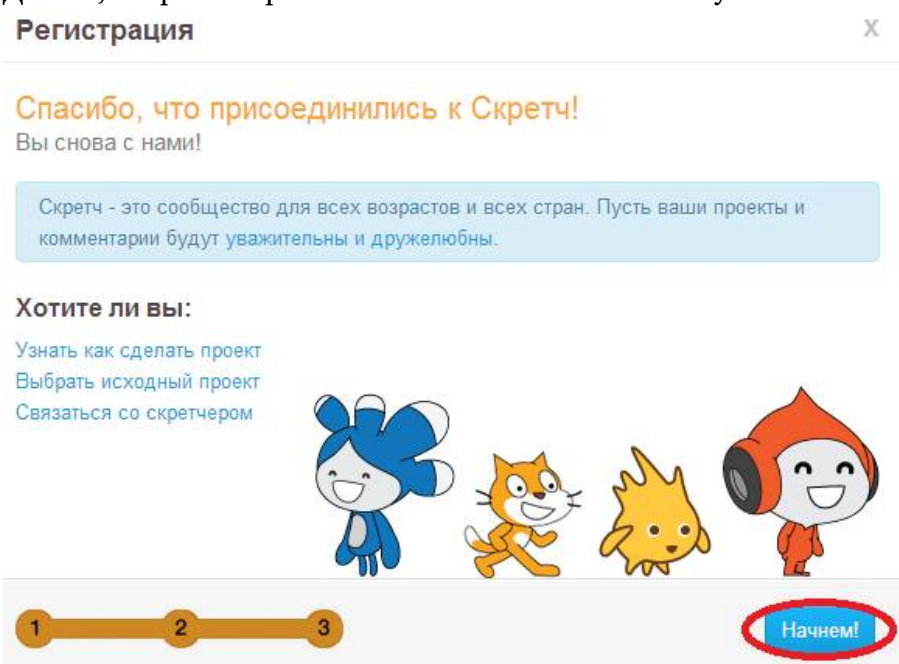



Затем надо заполнить информацию о себе. Здесь лучше написать правду.

Обратите внимание, если Вы прилетели с другой планеты, то кроме мужского и женского пола Вы можете выбрать любой пол и ввести его название, которое принято на Вашей планете.




После того, как Вы заполните информацию о себе и нажмёте «Далее», откроется финальное окно. Нажмите кнопку «Начнём!».

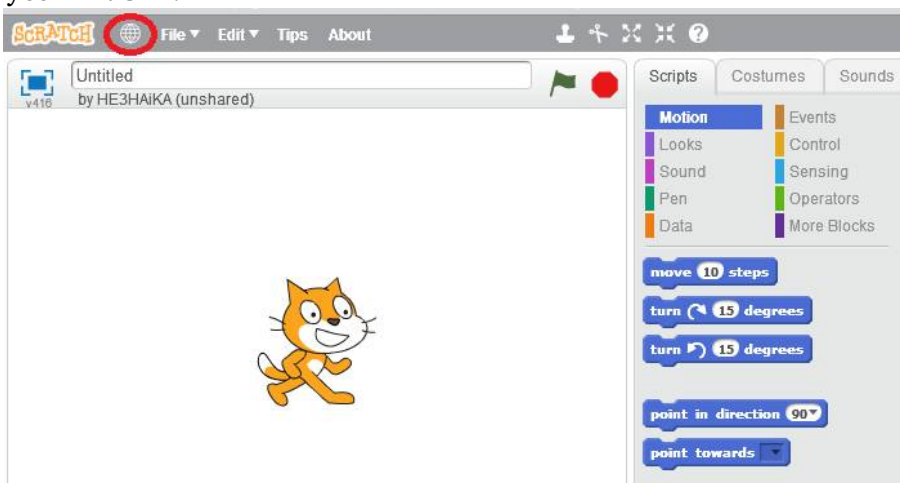



Поздравляю! Вы зарегистрировались! На верху справа Вы увидите Ваш логин и некий конвертик с цифрой один, которая означает, что в почтовом ящике уже есть одно письмо. Кликнув на конвертик, Вы можете прочитать сообщение.

Чтобы начать делать проекты кликните на папку с буквой «S».

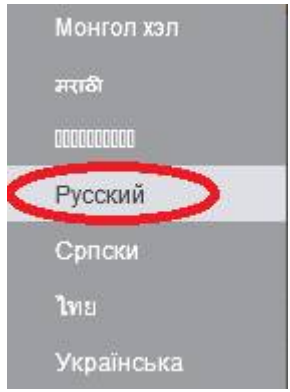
Вы попадёте на страницу «Мои работы». Нажмите на кнопку «+ Новый проект».

Откроется окно Scratch 2. Первым делом надо перейти на русский язык.




Нажмите на глобус, двигайте мышку вниз списка, на треугольничек, чтобы список начал прокручиваться.

Почти в самом низу списка будет русский язык.




Готово! Можно начинать создавать проекты.

КАК СКАЧАТЬ И УСТАНОВИТЬ ОФФЛАЙНОВЫЙ РЕДАКТОР SCRATCH.

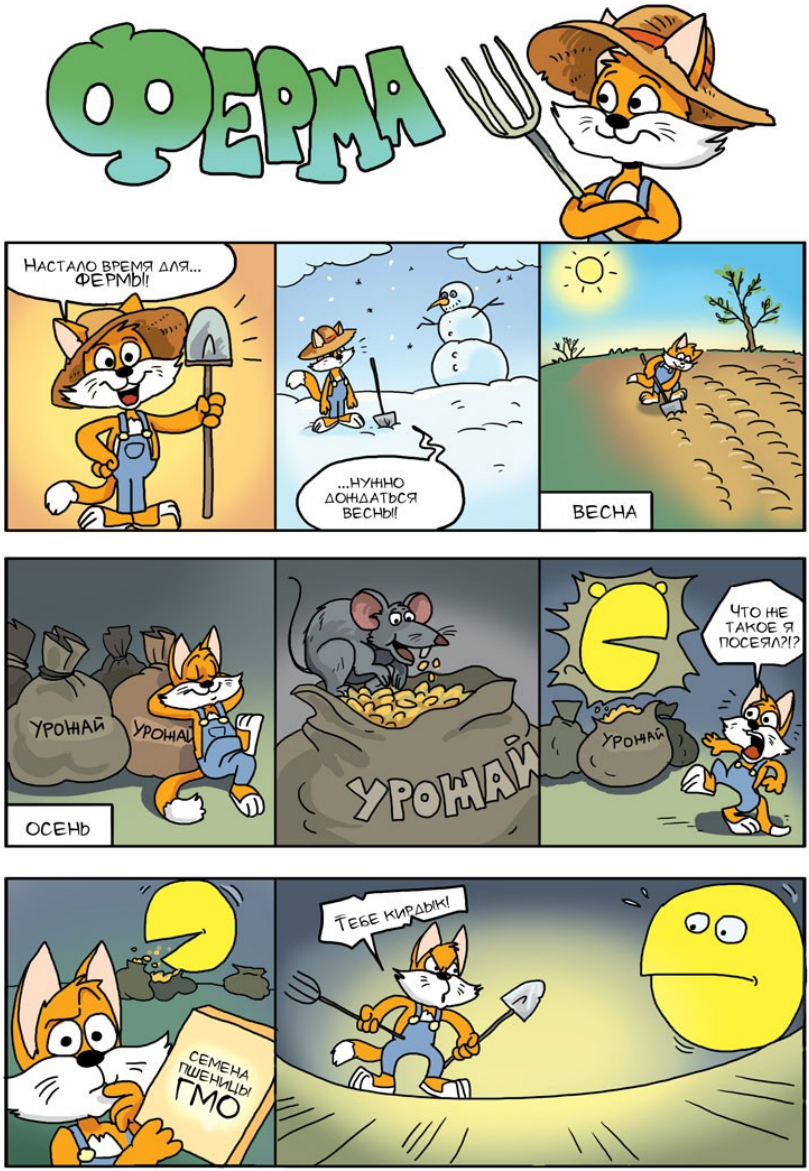
Это сделать очень просто. Перейдите по адресу http://scratch.mit.edu/scratch2download/. Если Ваш компьютер работает под управлением Windows, то кликайте туда, куда указывает красный овал.

Затем установите Scratch.

После установки на рабочем столе появится симпатичный ярлык.

Можно начинать!

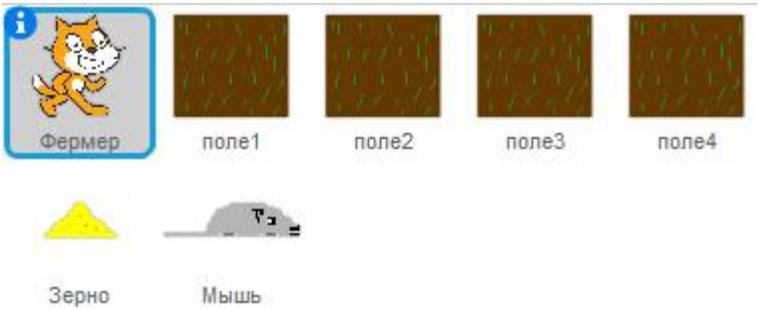
# ГЛАВА 1. ФЕРМА.




Решил Кот стать фермером и засеять четыре поля пшеницей. С тех полей, которые он засеял весной, осенью он собирал урожай. Но вот в амбаре завелась Мышь, и стала таскать зерно. Помогите Фермеру вырастить большой урожай! Количество зерна

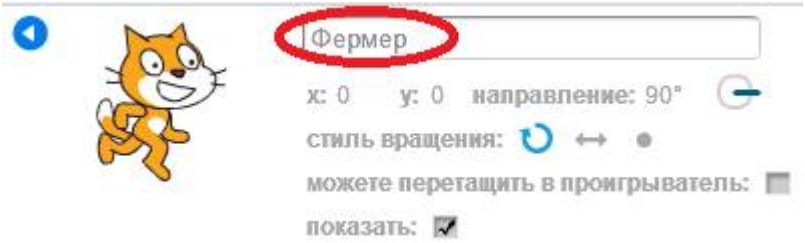
показано в верхнем левом углу экрана.

Всего у нас в игре будет 7 спрайтов. Фермер, Мышь, Зерно и четыре поля.

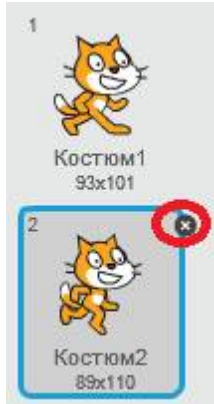



Создайте новый проект. Переименуйте Sprite1, для этого нажмите на синий кружочек.

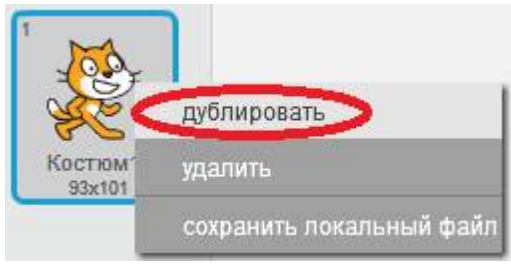
Введите имя «Фермер» и снова нажмите на синий кружочек, окошко свойств спрайта свернётся.




Перейдите на вкладку Костюмы и удалите второй костюм.




Нажмите правой кнопкой на Костюм1 и выберите «Дублировать».




Появится копия Костюма1. Подрисуйте на нём лопату или мотыгу. Это будет Фермер засевающий поле.

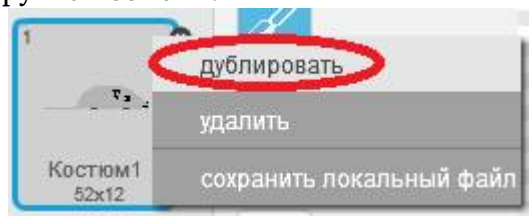
Затем нарисуйте Зерно. Кликните кисточку и нарисуйте маленькую кучку зерна.

После сбора урожая, когда зерна станет больше, размер кучки увеличится.

Теперь нарисуйте Мышку. У неё будет два костюма, один с открытым ртом, другой с закрытым.

Нарисуйте Мышку с закрытым ртом.

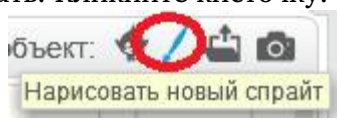
Продублируйте Костюм1.



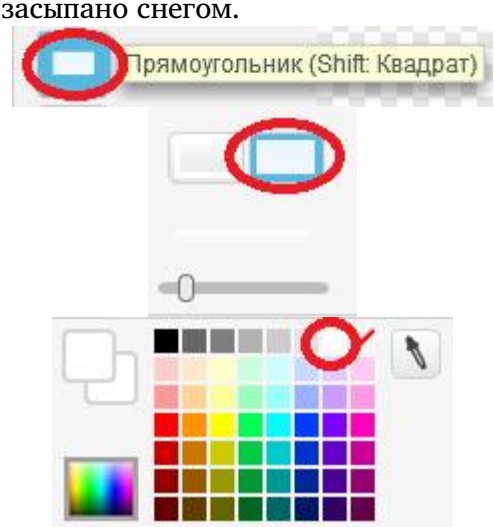

У Костюма2 сделайте рот Мышки открытым.

Теперь нарисуем спрайт первого поля. После того, как Вы сделаете ему программу и три раза продублируете, у Вас будет четыре одинаковых спрайта. Останется только немного их исправить. Всего у каждого спрайта поля будет по четыре костюма. Первый костюм — белое зимнее поле, второй костюм — коричневое весеннее поле, с первой травкой. Третье поле — летнее, зелёное, а четвёртое, снова коричневое с жёлтой пожухлой травой. Выглядеть спрайты полей будут так.

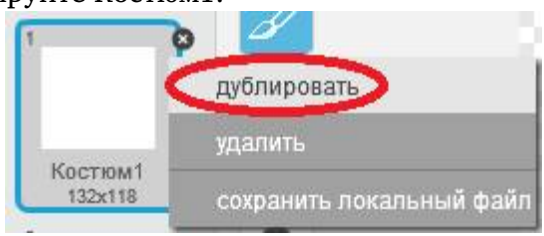
Начинаем рисовать. Кликните кисточку.



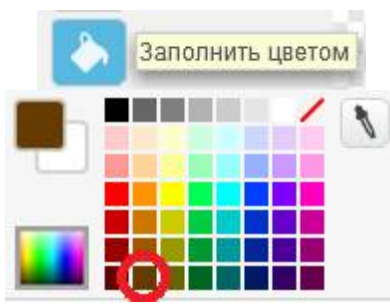

Нарисуйте белый квадрат. Это будет первое время года — зима. Всё поле засыпано снегом.

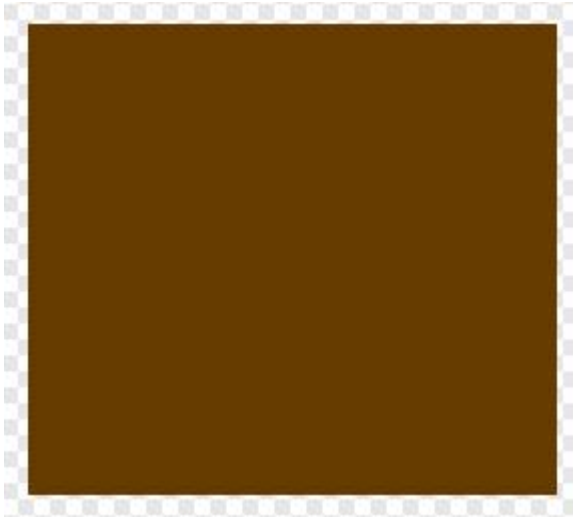


Продублируйте Костюм1.



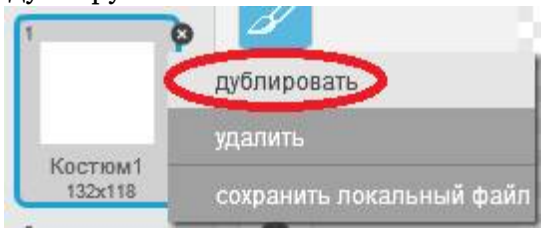

Затем нарисуем второе время года — весну. Залейте Костюм2 коричневым.



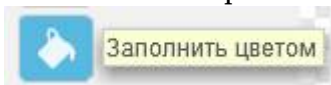


И подрисуйте немного весенней травки.

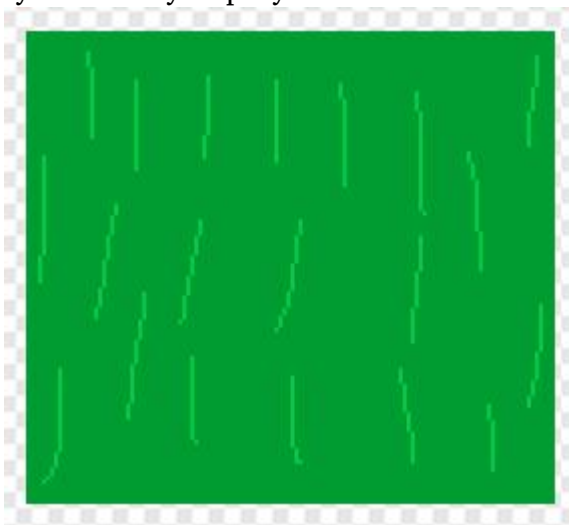
Снова продублируйте Костюм1.



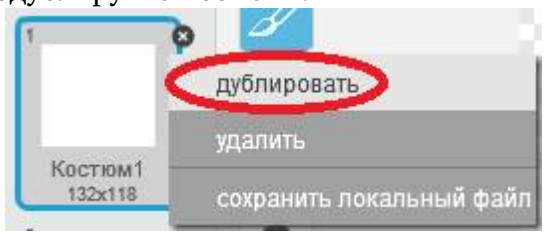

Костюм3 будет летним. Залейте фон зелененьким.



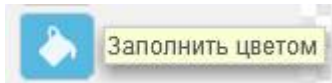
И подрисуйте высокую траву.



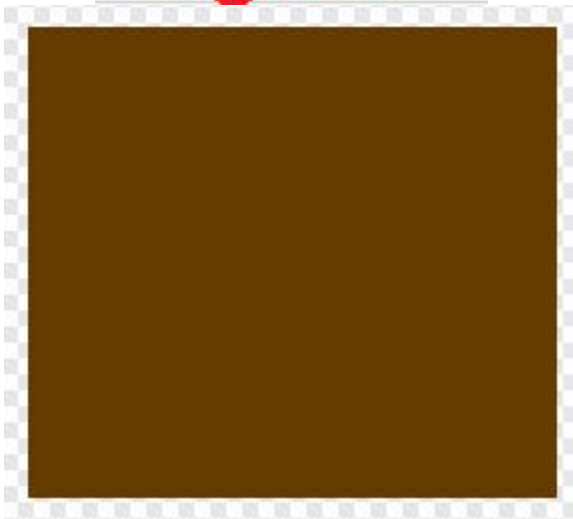
Снова продублируйте Костюм1.




Четвёртый костюм будет осенним. Трава пожухла и лежит на земле. Залейте

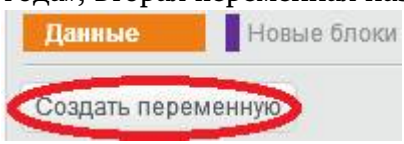


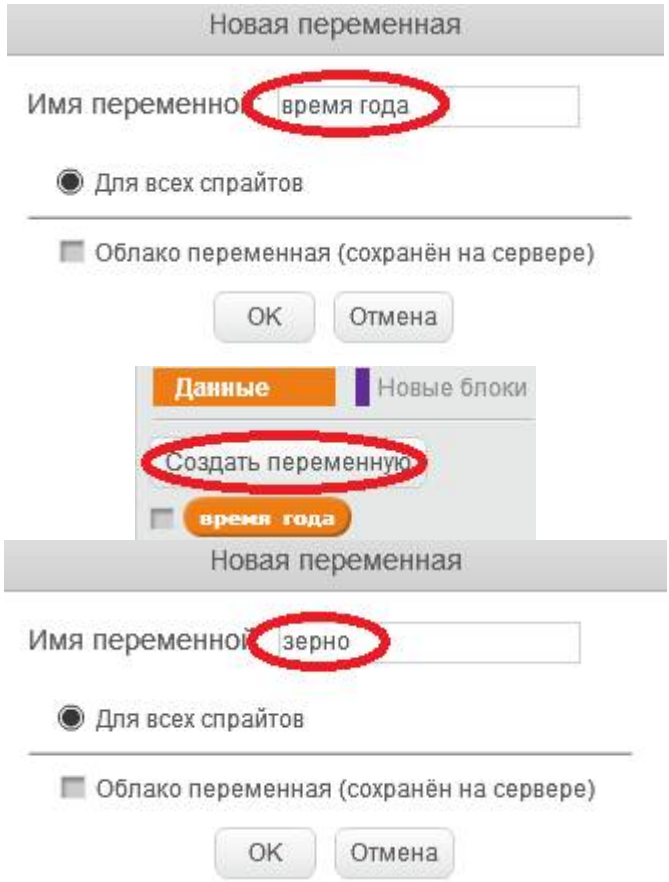

Костюм4 коричневым.



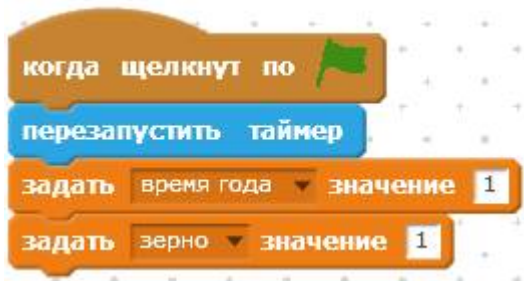
И нарисуйте желтую пожухшую траву.

Все спрайты готовы. Можно делать программу для Сцены, но сначала надо создать две переменные. Первая переменная называется «время года», вторая переменная называется «зерно».






Программа Сцены состоит из двух скриптов. Первый скрипт перезапускает таймер и задаёт начальные значения переменным.



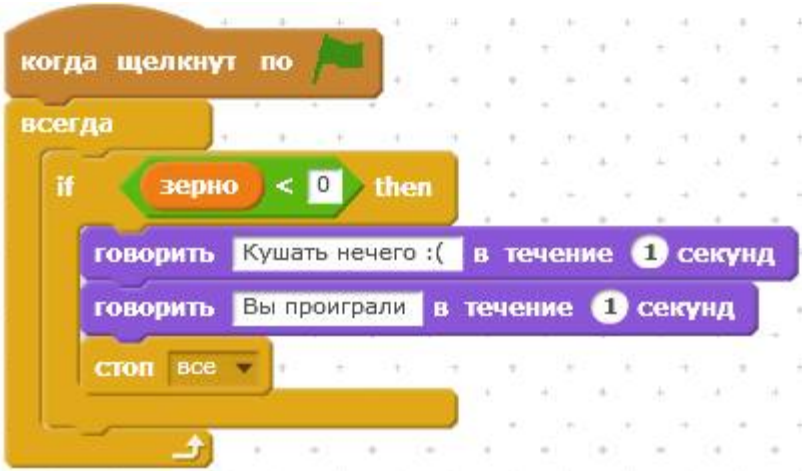

Каждое время года длится по пять секунд. Значение блока таймер увеличивается каждую секунду на единицу. Если прошло больше пяти секунд, то значение переменной «время года» увеличится на единицу. Когда зима, значение переменной «время года» равно единице, весна = 2, лето = 3, осень = 4. Если значение этой переменной превысит четыре, то значение переменной вернётся к единице — настанет Зима. В этот момент количество зерна уменьшится на одну порцию — это Фермер у нас наелся на год вперёд. Число -1 характеризует аппетит Фермера, если вы

измените его на -2, то Фермер будет съедать по две порции зерна за год и играть будет сложнее.

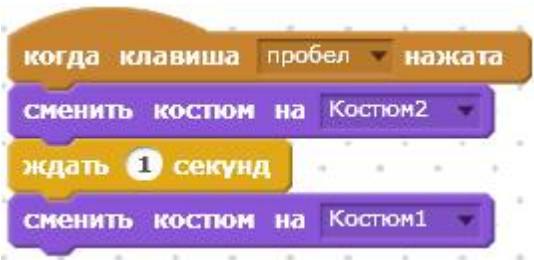
Теперь сделаем программу Фермеру. Она состоит из семи скриптов. Первый скрипт задаёт начальное положение и размер Фермера. Если не использовать блок «перейти в верхний слой», то, иногда, Фермер бегает позади поля и его не видно.

когда щелкнут по
перейти в x: 100 y: 100
установить размер 50 %
сменить костюм на Костюм1
перейти в верхний слой
говорить Привет, я фермер. в течение 1 секунд
говорить Сей зерно весной нажимая пробел. в течение 1 секунд

Скрипт конца игры. Если зерно закончилось, то всё.


когда щелкнут по
всегда
if зерно < 0 then
говорить Кушать нечего :( в течение 1 секунд
говорить Вы проиграли в течение 1 секунд
стоп все

Скрипт посева зерна. Фермер на одну секунду будет менять костюм — копать лопатой.




Четыре скрипта управления движением Фермера.

когда клавиша стрелка направо нажата
изменить x на 5
когда клавиша стрелка влево нажата
изменить x на -5
когда клавиша стрелка вверх нажата
изменить y на 5
когда клавиша стрелка вниз нажата
изменить y на -5

Затем сделаем программу Зерну. С каждым зёрнышком размер кучи увеличивается на 10%.

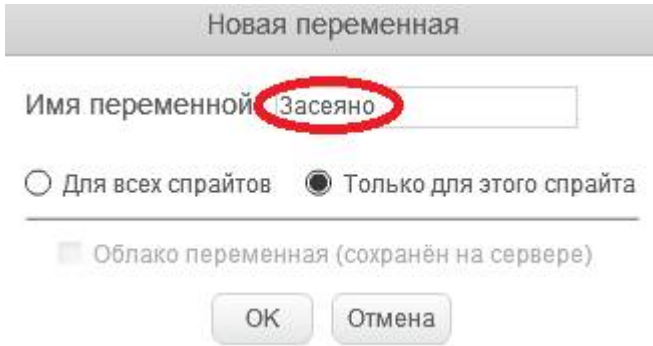
Скрипт Мыши. В начале работы программы Мышь не видно.

Мышь съедает очень много зерна. После мышиного обеда количество зерна уменьшается в два или в три раза.

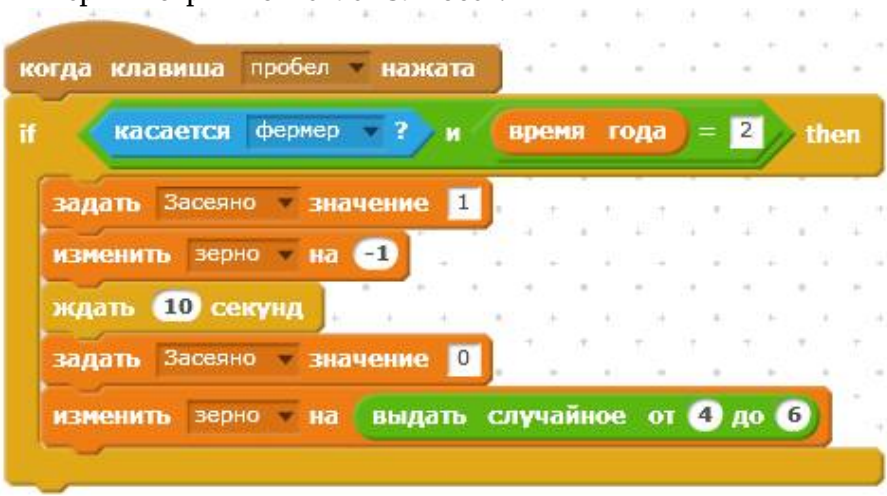
На обед Мышь выбегает через пять-десять секунд. Ох, и голодная Мышка!

когда щелкнут по
ждать 30 секунд
всегда
ждать выдать случайное от 5 до 10 секунд
показать
плыть 1 секунд в точку x: 170 y: -160
задать зерно значение округлить зерно / выдать случайное от 2 до 3
скрыть
перейти в x: 60 y: -160

Настало время запрограммировать первое поле. Для начала создайте локальную переменную «Засеяно». Она будет принимать значение 1, если поле засеяно и 0, если засеять не успели.




Первый скрипт отвечает за посев.


когда клавиша пробел нажата
if касается фермер ? и время года = 2 then
задать Засеяно значение 1
изменить зерно на -1
ждать 10 секунд
задать Засеяно значение 0
изменить зерно на выдать случайное от 4 до 6

Следующий скрипт отвечает за смену времён года. Поле изменяет свой костюм в соответствии со значением переменной «время года», и только, если поле не засеяно, лето не наступает — пшеница не вырастает, урожая не будет. Очень печально.

когда щелкнут по
задать Засеяно значение 0
перейти в x: -170 y: -120
всегда
if Засеяно = 1 then
сменить костюм на время года
иначе
if время года = 1 then
сменить костюм на Костюм1
if время года = 2 then
сменить костюм на Костюм2
if время года = 3 then
сменить костюм на Костюм2
if время года = 4 then
сменить костюм на Костюм4

Теперь три раза продублируйте поле1. У полученных полей надо изменить только значения X и Y в блоке «идти в X Y». Для скрипта поля2 измените значения так.

Для поля3 сделайте так.

Для поля4 сделайте так.

Вопросы по игре «Фермер».

1. Как изменить скорость передвижения Фермера?

2. Как увеличить урожай, чтобы в конце года появлялось больше зерна?

3. Как увеличить продолжительность года до одной минуты?

Задания к игре «Фермер».

1. Увеличьте начальное количество Зерна, чтобы Фермеру было легче жить.

2. Увеличьте аппетит Фермера. Пускай съедает по одному Зерну каждый сезон.

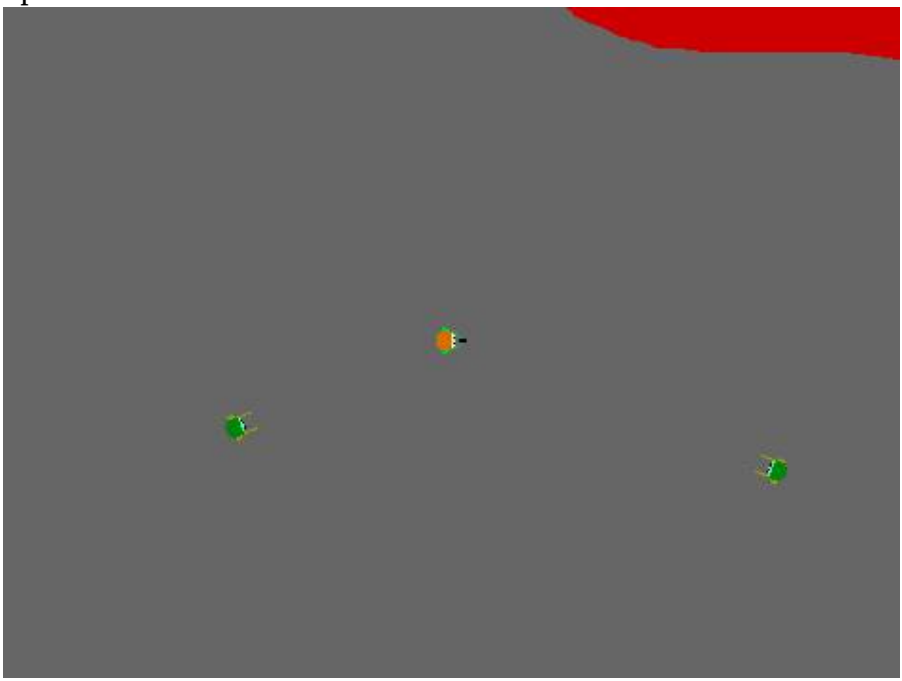
3. Измените количество зерна, съедаемого Мышкой за раз, чтобы Зерно не кончалось слишком быстро.

4. Добавьте Фермеру возможность прогнать Мышь. Если Фермер стукнет её лопатой, то Мышь целый год, 20 секунд, не появляется — лечится.

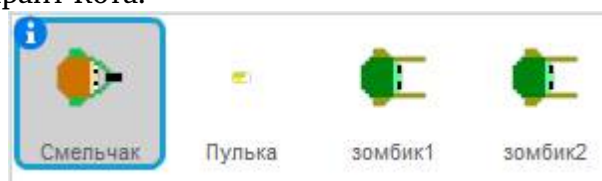
# ГЛАВА 2. АТАКА ЗОМБИ.

Как-то раз, один Смельчак решил отправиться в долину зомби, чтобы покончить с ними раз и навсегда. У него было оружие с большим запасом патронов и храброе сердце. Удастся ли ему победить в этой схватке? Об этом Вы узнаете из нашей

игры.

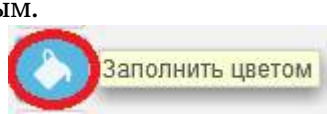


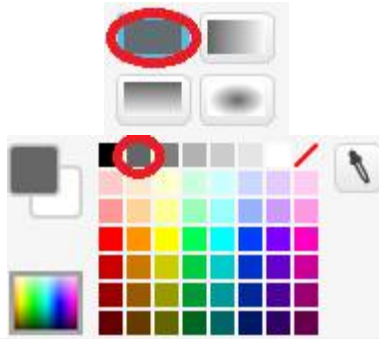
В этой игре у нас будет всего четыре спрайта: Смельчак, его верная Пуля, и два злобных зомби. Смотреть на них мы будем сверху. Как вы поняли из комикса, котиков в этой игре не будет… Удалите спрайт Кота.




Сначала, как обычно нарисуем фон и спрайты героев игры. Фон можно сделать как у меня — однотонным, но можно и усложнить игру, нарисовав на фоне какие-нибудь кусты и коряги. Выберите сцену.

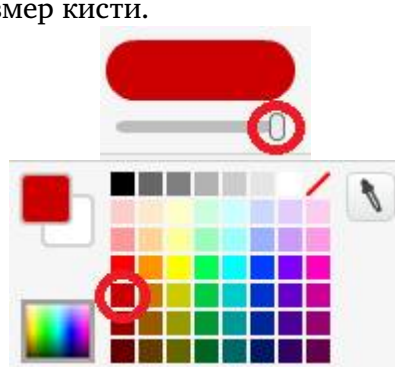
Залейте фон серым.





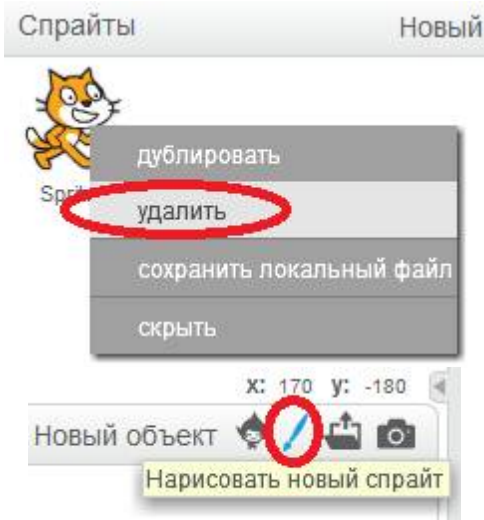
Выберите инструмент Кисть.

Выберите красный цвет, и нарисуйте лаву. Установите максимальный размер кисти.



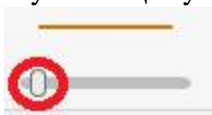
Нарисуйте в углу озеро лавы.

Фон готов. Теперь удалите вездесущего Кота, и нарисуйте Смельчака.




Выберите Кисть.

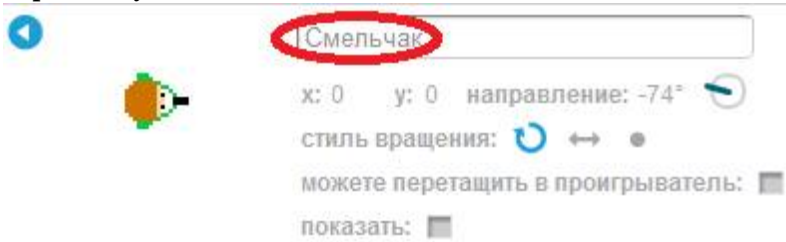
Установите минимальную толщину Кисти.



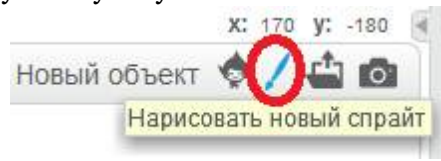
Масштаб сделайте очень крупным.

Выбирая разные цвета, тонкой Кистью нарисуйте Смельчака, как будто смотрите на него сверху.

Переименуйте Смельчака.




Потом нарисуйте Пульку.

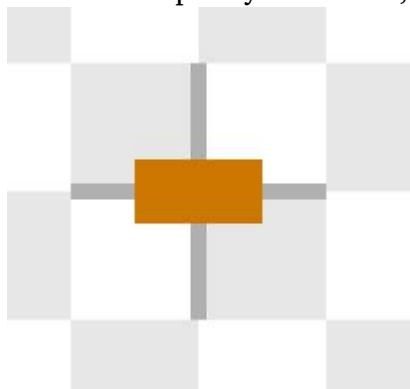



Выберите Кисть.

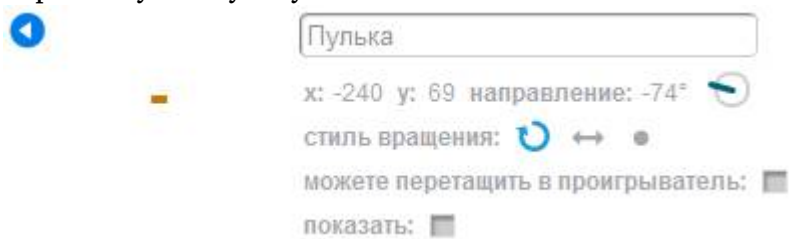
Установите минимальную толщину Кисти.

Масштаб сделайте очень крупным.

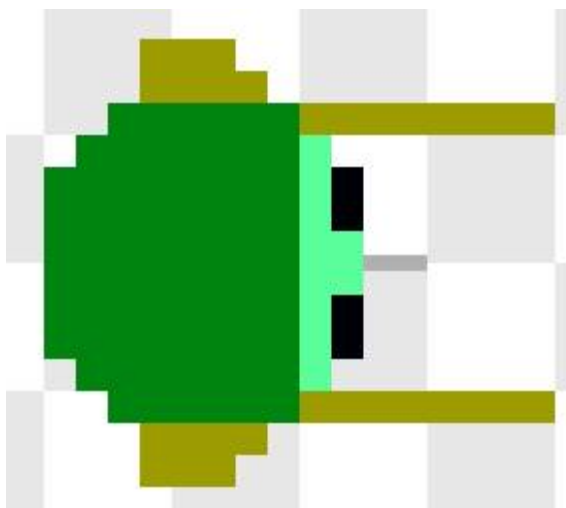
Нарисуйте малюсенький прямоугольничек, это будет Пулька.



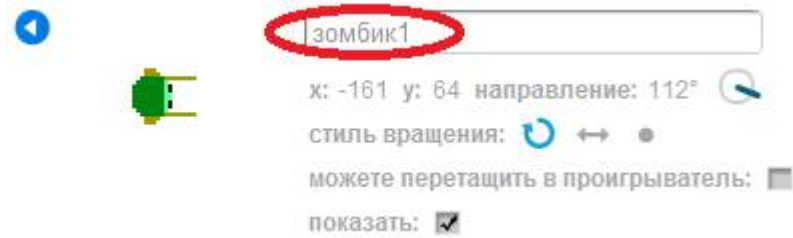
Переименуйте Пульку.




Аналогичным образом нарисуйте первого Зомби. Выберите тоненькую Кисть и увеличьте масштаб, чтобы было удобнее рисовать.

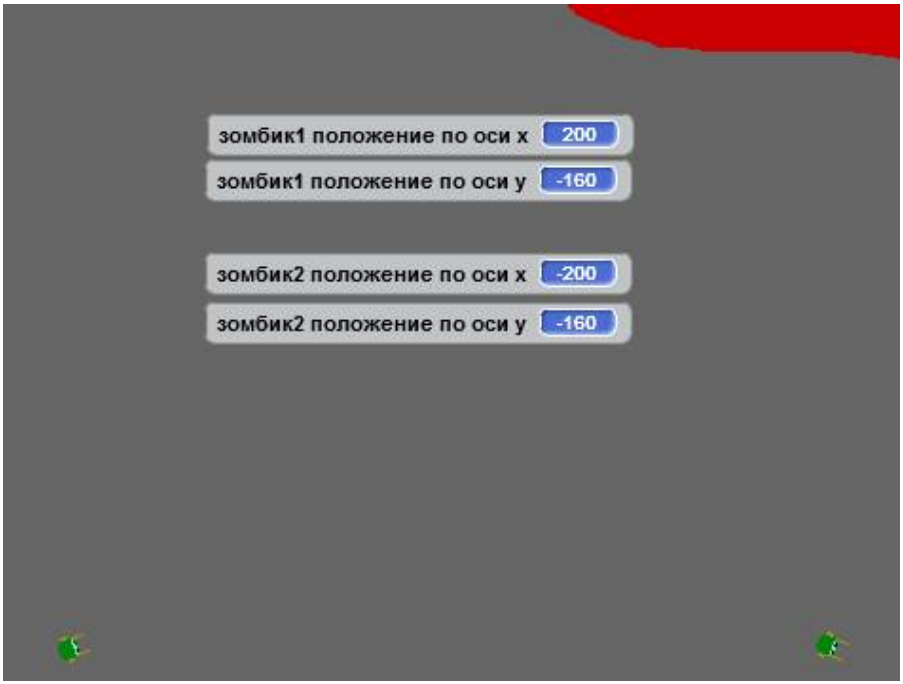


Переименуйте первого Зомби.

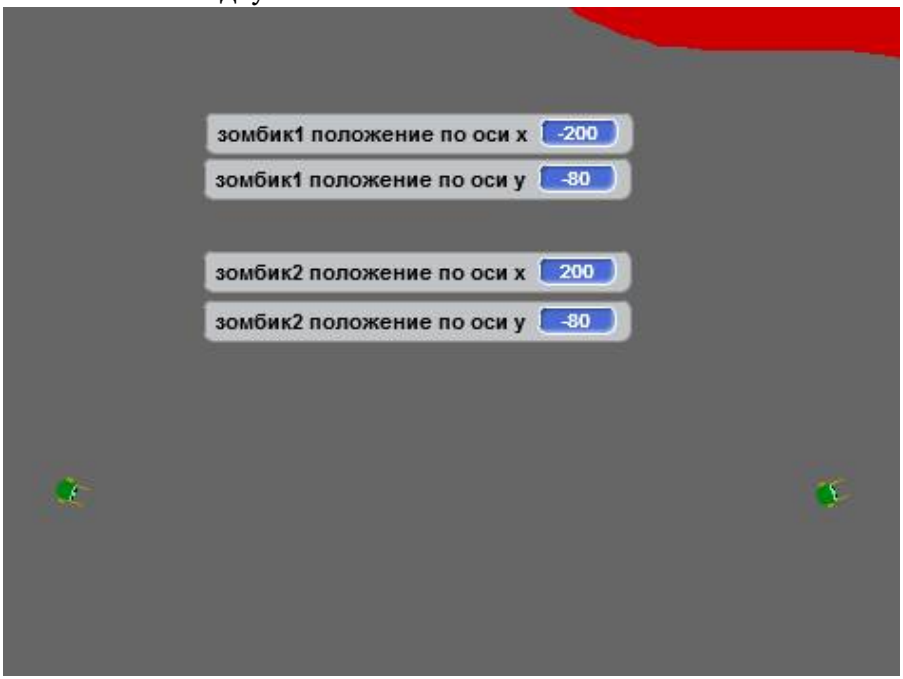



Второго Зомбика пока рисовать не будем. После того, как запрограммируем поведение первого, то просто продублируем его — получится второй Зомбик.

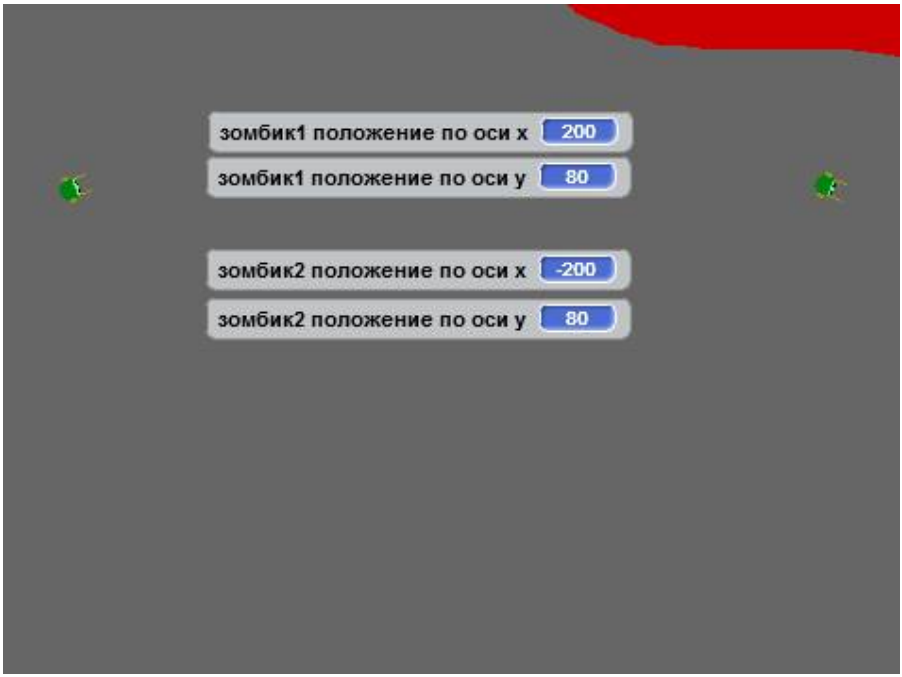
Начинаем делать программу. Сначала создадим два списка. Список, это такой объект, который содержит в себе несколько переменных. Нам список понадобился для того, чтобы задать места появления Зомбиков. Если они будут появляться в любом месте экрана, то будет очень трудно играть, ведь Зомбик может появиться очень близко к отважному парню, и сразу его слопать. Для того, чтобы Зомбики появлялись только по краям экрана, мы зададим конкретные места их появления. Для этого нам и понадобятся списки. В первом списке будут лежать координаты Y мест появления Зомбиков, во втором списке будут храниться координаты X их мест появления. В списке «места Y» будут лежать четыре значения: -160, -80, 80, 160. В списке «места X» будут храниться два значения: -200 и 200. Задав такие значения координатам мест появления Зомбиков, мы получим восемь возможных мест их появления. Каждый из Зомбиков может появится либо в этих двух точках.


зомбик1 положение по оси x 200
зомбик1 положение по оси y -160
зомбик2 положение по оси x -200
зомбик2 положение по оси y -160

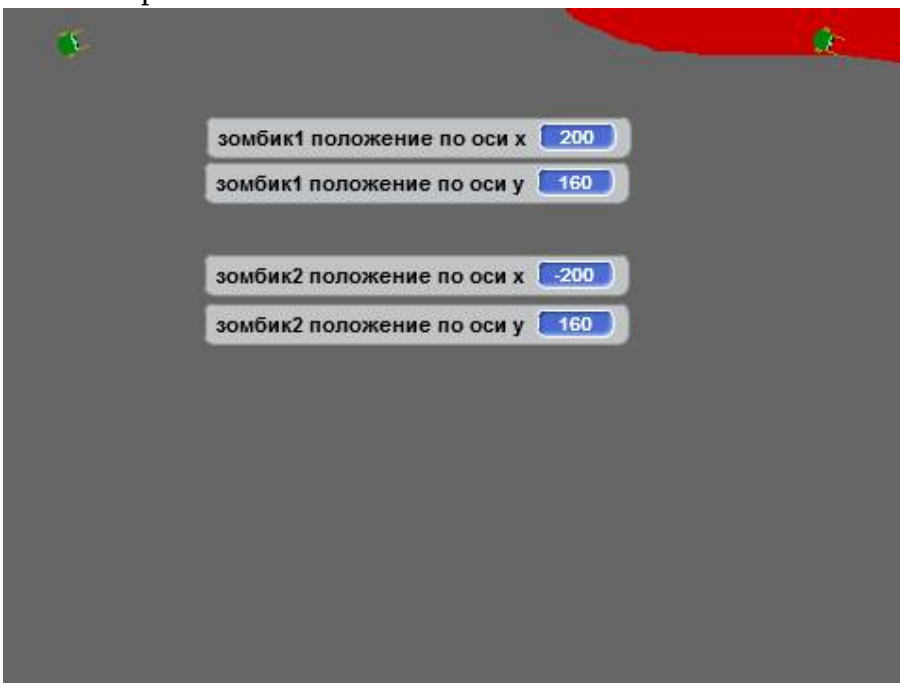
Либо в этих двух точках.




Либо в этих двух точках.

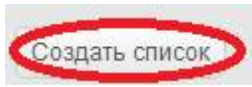



Либо в этих двух точках. Ничего, что один из Зомбиков появился прямо в лаве. Эти зомби лавы не боятся.

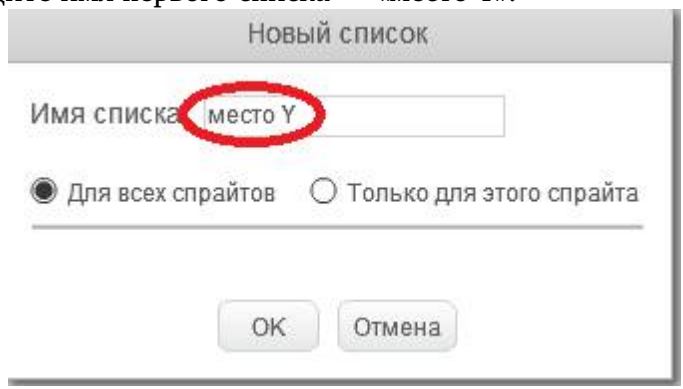

зомбик1 положение по оси x 200
зомбик1 положение по оси y 160
зомбик2 положение по оси x -200
зомбик2 положение по оси y 160

Вот как выглядят все эти восемь мест появления на одном рисунке.

Для того, чтобы создать списки, в оранжевых блоках нажмите «Создать список».




Введите имя первого списка — «место Y».




Появится пустой список. Поставьте около него галочку, он отобразится на экране. Перетащите его повыше, чтобы он был виден целиком. Сейчас список пустой. Внизу указано, что длина списка (length) равна нулю.

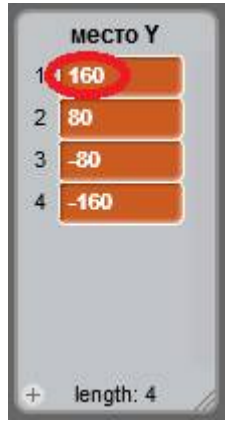
Давайте заполним его необходимыми значениями. Нажмите внизу плюсик.

Появится первый элемент списка с неопределённым значением.

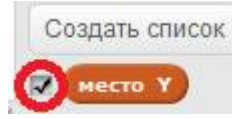
Введите туда число -160.

Снова нажмите плюсик, и введите число -80.

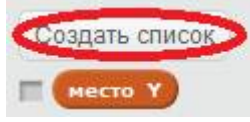
Введите ещё два значения, 80 и 160. Список примет следующий вид. Обратите внимание, длина списка (length) стала равна четырём.



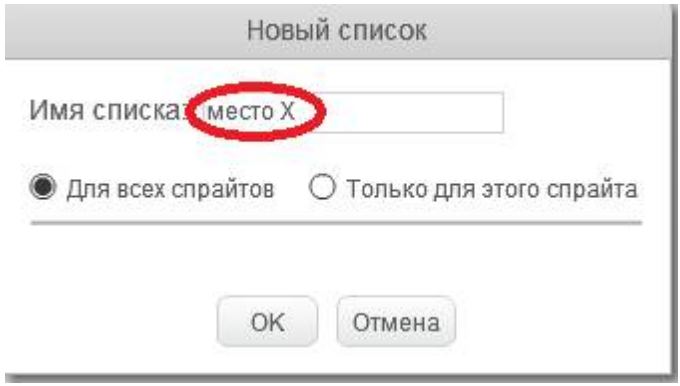

Снимите галочку около списка, он пропадёт с экрана.




Теперь создадим второй список. Нажмите «Создать список».




Введите название.




Перетащите список повыше, и заполните значениями.

Введите значения -200 и 200.

Снимите галочку около списка, он пропадёт с экрана.

Теперь, когда списки готовы, можно начинать программировать спрайты. Сначала сделайте программу для Смельчака. В его программе только один скрипт, зато большой. Управляется наш камикадзе стрелками. Вверх и вниз — бег, вправо и влево — поворот. Пробел передаёт событие «стреляю». На это событие будет реагировать Пулька.

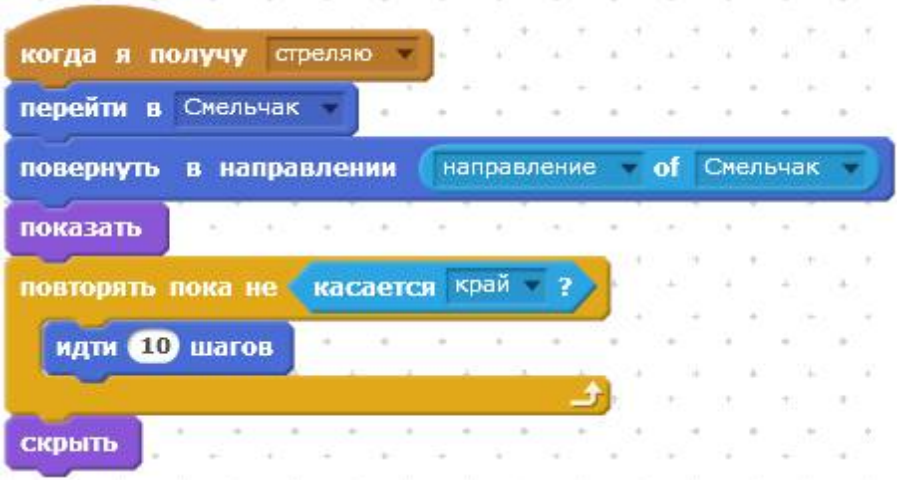
Если «герой, штаны с дырой», коснётся хотя бы одного из Зомбиков - «давай, до свиданья».

Программа Смельчака.

когда щелкнут по
перейти в x: 0 y: 0
показать
всегда
если клавиша стрелка вверх нажата? то
идти 3 шагов
если клавиша стрелка влево нажата? то
повернуть на 7 градусов
если клавиша стрелка вниз нажата? то
идти -3 шагов
если клавиша стрелка направо нажата? то
повернуть на 7 градусов
если клавиша пробел нажата? то
передать стреляю
если касается зомбик1 ? или касается зомбик2 ? то
думать Не повезло... 1 секунд
скрыть
стоп все

У Пульки всего два скрипта. Первый скрипт просто её прячет.

Второй скрипт отрабатывает событие «стреляю». Сначала Пулька перемещается в центр Смельчака, поворачивается в том же направлении, куда смотрит Смельчак, а потом летит до края и исчезает.



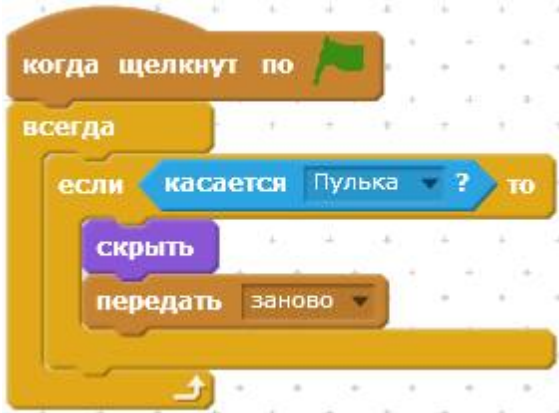

Теперь сделаем программу зомбику1. Всего у него будет четыре скрипта. Первый скрипт прячет зомбика1, ждёт от одной секунды до трёх и устанавливает зомбика1 в одно из восьми мест. Когда зомбик установлен в нужное место, он показывается, и начинает движение, пока не поймает Смельчака, или не словит Пулю.

когда щелкнут по
скрыть
перейти в x: 0 y: 0
ждать выдать случайное от 1 до 3 секунд
установить x в элемент случайное из место X
установить y в элемент случайное из место Y
показать
повторять пока не касается Пулька ? или касается Смельчак ?
идти 1 шагов

Второй скрипт устанавливает направление движения

зомбика1. Он всегда двигается по направлению с нашему герою.

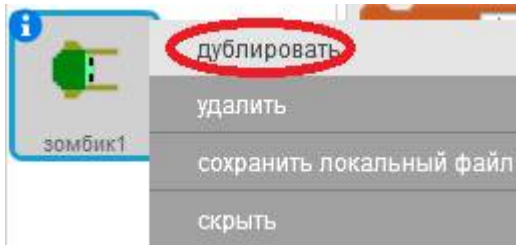
Третий скрипт определяет взаимодействие с Пулькой. Касаясь её, зомбик1 исчезает, на прощание передавая событие «заново». Это событие сразу же порождает нового монстра, который незамедлительно атакует Смельчака.



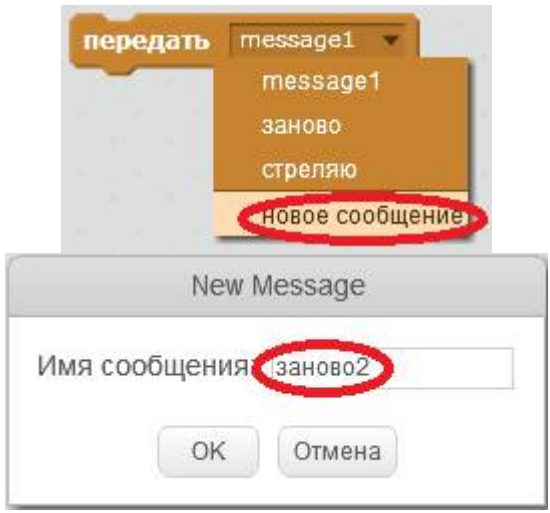

Четвёртый скрипт отвечает за обработку события «заново». Новый зомбик помещается в одну из восьми начальных точек, появляется, и как обычно бегает за смелым героем, ни на секунду не давая ему передышки.

когда я получу заново
установить x в элемент случайное из место X
установить y в элемент случайное из место Y
показать
повторять пока не касается Пулька ? или касается Смельчак ?
идти 1 шагов

Все четыре скрипта зомбика1 готовы. Дублируете этого товарища.




У него автоматически появится имя зомбик2. Все его скрипты полностью идентичны скриптам зомбика1, за исключением передаваемого события. Если первый зомбик передавал событие «заново», то второй будет передавать событие «заново2». Создайте новое событие «заново2».




Замените в скриптах зомбика2 все события «заново» на «заново2». Если этого не сделать, то при гибели любого из двух зомби возрождаться будут сразу оба, а не только тот, который погиб.

Игра готова. Попытайтесь продержаться как можно дольше.

Вопросы по игре «Атака зомби».

1. Как ускорить движение зомбиков?
2. Как ускорить движение Смельчака?
3. Как ускорить движение Пули? Какой баг может появиться, если движение пульки очень быстрое?

Задания по доработке игры «Атака зомби».

1. Сделайте так, чтобы направление движения Смельчака

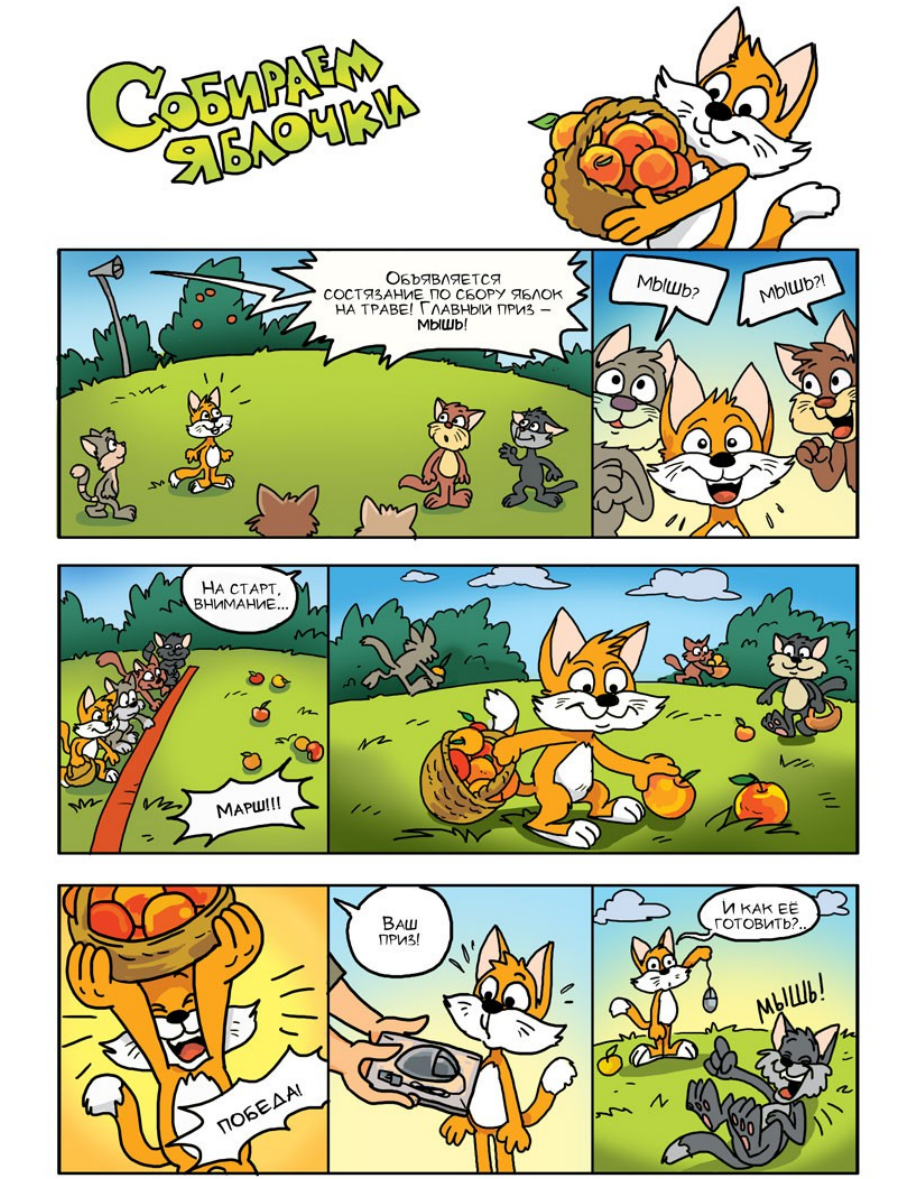
задавалось с помощью мышки.

2. Сделайте третьего зомбика.

3. Увеличьте до десяти количество мест, в которых может появиться зомби.

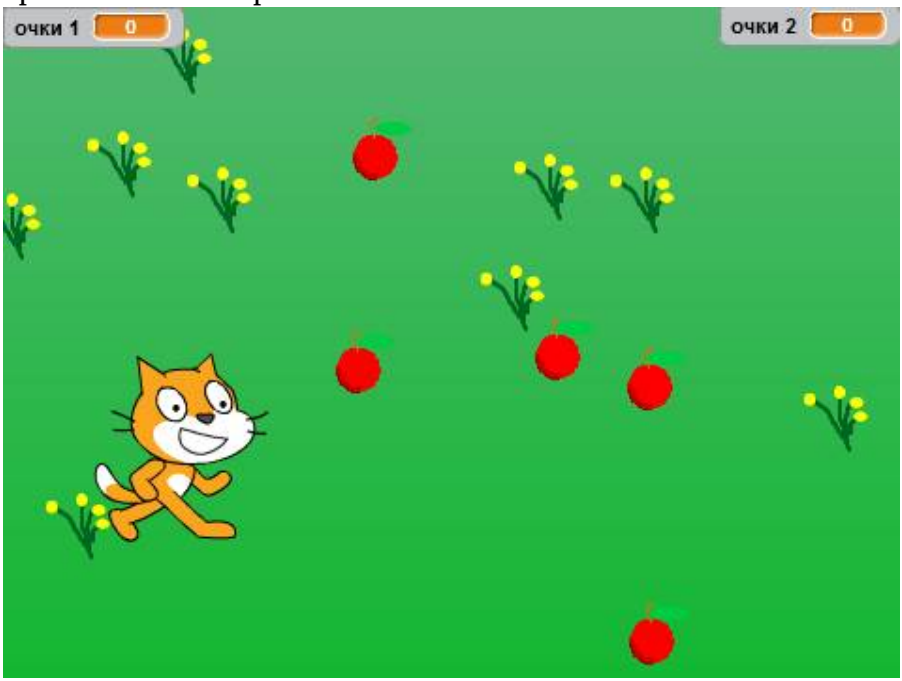
4. Сделайте Смельчаку 5 жизней.

# ГЛАВА 3. СОБИРАЕМ ЯБЛОЧКИ.



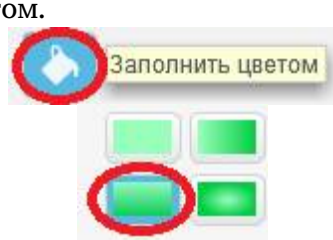

Сделаем игру про двух котов. В неё можно будет играть вдвоём. На поле высыпаются 20 яблочек. Размер поля больше экрана, поэтому видны не все яблоки. Сначала, в течение 20 секунд яблоки собирает первый игрок, он старается собрать как

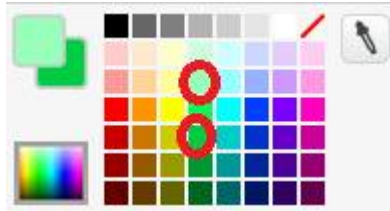
можно больше, потом, яблоки рассыпаются снова, и к сбору урожая приступает второй кот. Кто больше соберёт, тот скажет: «Я выиграл!». В процессе игры Коты стоят неподвижно, а экран с яблоками и цветочками под ним прокручивается по вертикали и по горизонтали. Этот эффект называется вертикальный и горизонтальный скроллинг.



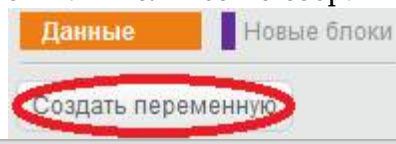

Сперва нарисуем фон и все спрайты. Фон просто зальём зелёным цветом с небольшим градиентом. Потом раскрасим и запрограммируем первого кота — Ваську. Потом сделаем второго кота — Рыжика, его раскрашивать не надо — он и так рыжий! После этого нарисуем и запрограммируем яблоко. Да-да, у нас будет только одно яблоко, а то, что вы видите на экране — это его клоны, о них потом поговорим подробнее. Когда яблоко будет готово, из него с помощью дублирования сделаем цветочки. Они будут расти на поле для красоты. С ними движение котов будет выглядеть реалистичней.

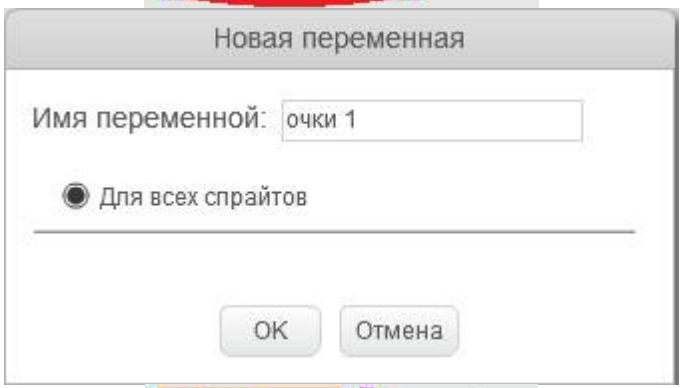
Начинаем с зелёного фона. Залейте фон зелёным цветом с небольшим градиентом.

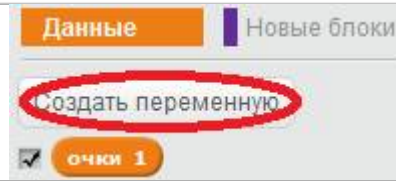


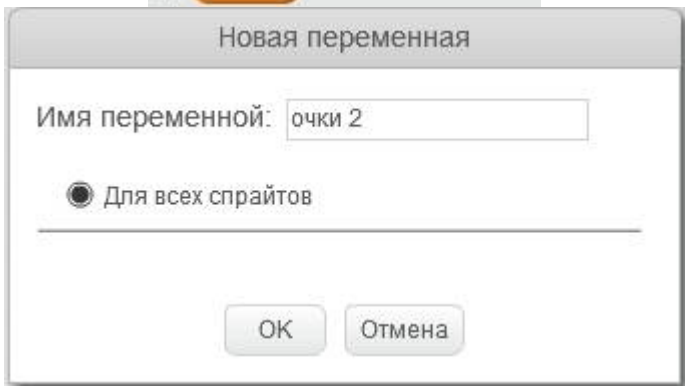


Сцене сделайте маленькую программу по обнулению переменных. Создайте две переменные «очки 1» и «очки 2». Эти переменные будут считать количество собранных яблок.

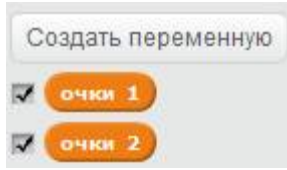


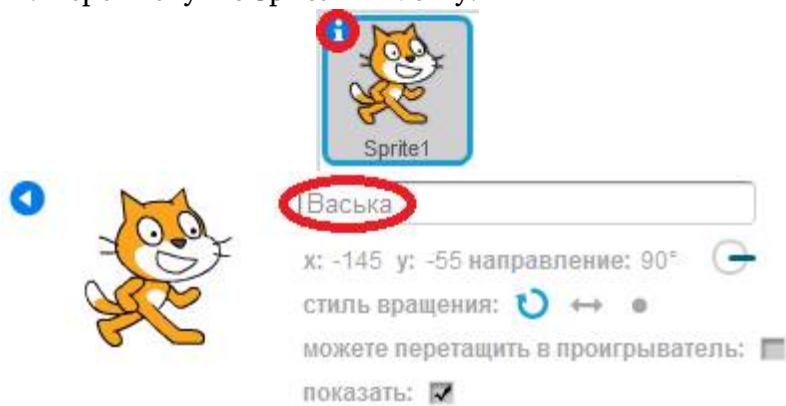




Получились две переменные.

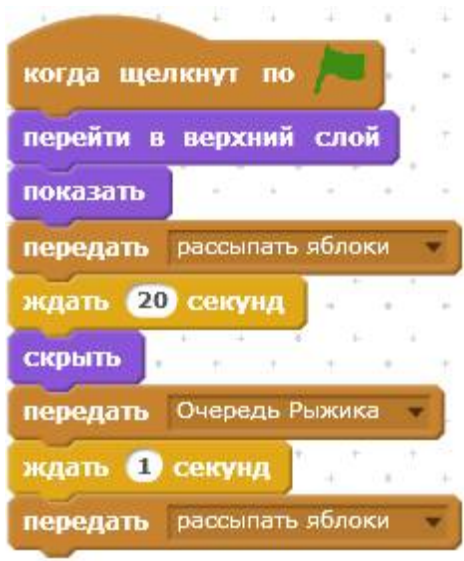



Сделайте для сцены такую программу.

Теперь приступаем к программированию поведения кота Васьки. Переименуйте Sprite1 в Ваську.

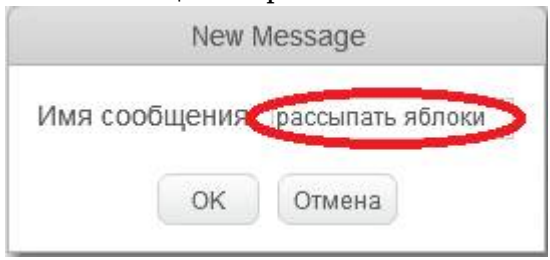



Сделайте ему простенькую программу из двух скриптов. Первый скрипт.




Не забудьте создать новое сообщение.

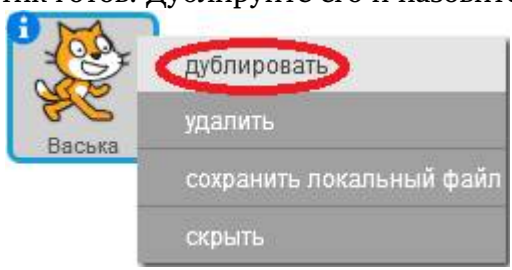
Назовите новое сообщение «рассыпать яблоки».

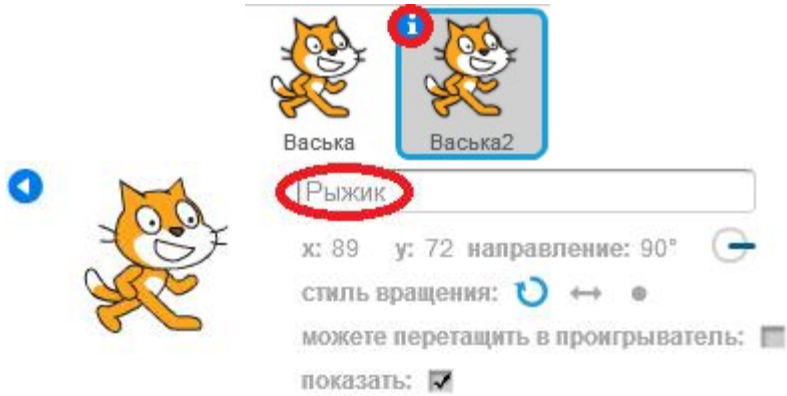



Создав первое событие, прикрепите к скрипту блок «передать «рассыпать яблоки». Второе событие создавайте только после того, как прикрепите первое событие с скрипту.

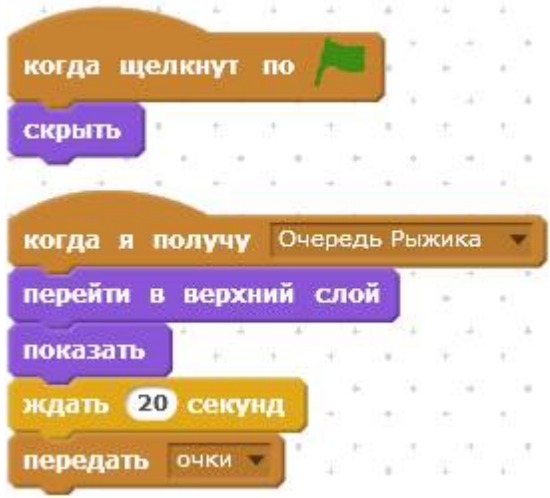
Когда первый скрипт будет готов, сделайте второй. Он будет работать в конце игры, при получении сообщения «очки». Если Васька наберёт больше яблок, чем Рыжик, то переменная «очки 1» будет больше переменной «очки 2», и Васька на весь двор объявит: «Я выиграл!». После этого работа программы будет остановлена блоком «стоп всё».

Первый котик готов. Дублируйте его и назовите Рыжиком.

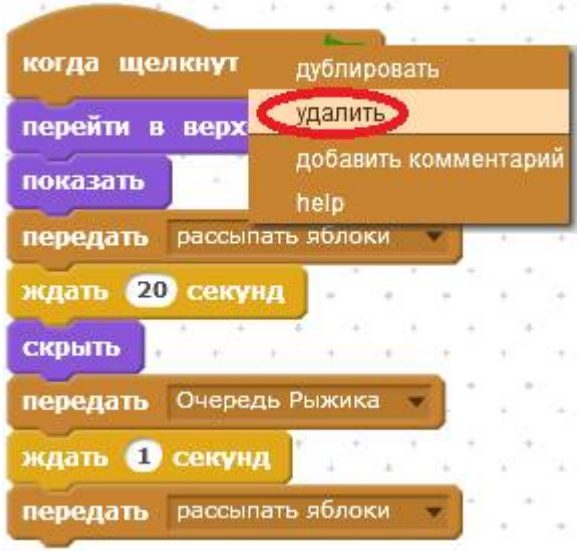





Сейчас программа Рыжика такая же, как и у Васьки. Надо немного её изменить. Сначала удалите первый скрипт.



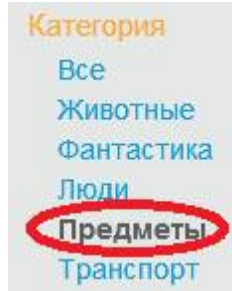

Вместо него создайте два маленьких скриптика. Первый скриптик скрывает Рыжика в начале игры, так как сначала очередь Васьки. Через двадцать секунд, когда миссия Васьки будет завершена, Рыжик получит сообщение «Очередь Рыжика» и приступит к сбору урожая. Ему тоже отведено двадцать секунд. После этого он передаёт сообщение «очки» и происходит подсчёт очков. Котик, который собрал больше яблок, скажет победные слова.




Третий скрипт Рыжика очень похож на такой же скрипт Васьки, только переменные надо переставить местами, и выкинуть блок «показать», ведь при подсчёте очков Рыжик и так виден.

Он только недавно закончил собирать яблоки, и ещё не успел никуда удрать.

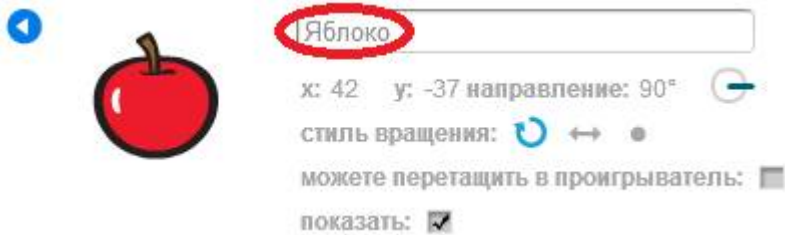
Коты к соревнованию готовы. Осталось сделать яблочко и цветочек. Начнём с яблока, потом его дублируем, и переделаем в цветочек. Программы-то у них почти одинаковые. Добавьте из библиотеки спрайтов яблоко.




Дважды кликните по яблоку.

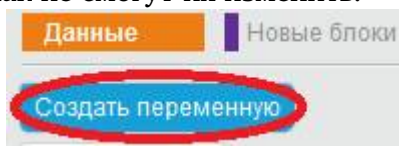
Apple

Переименуйте Apple в Яблоко.

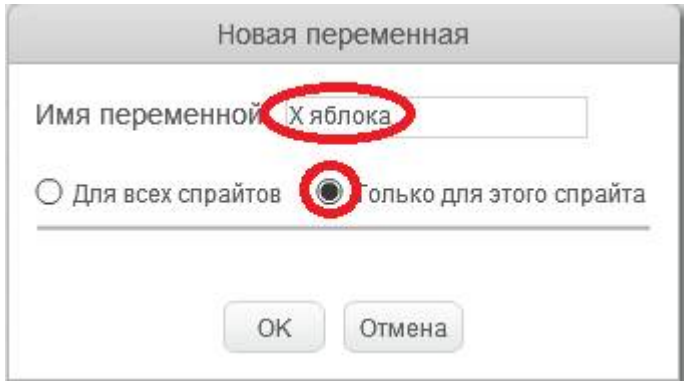



Программа у яблока очень сложная. В ней шесть скриптов, и ни один из них не начинается с блока с зелёным флагом. Сначала нам необходимо создать две локальные переменные «X яблока», и «Y яблока».

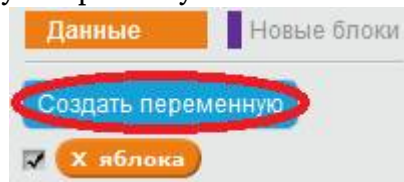
Эти переменные будут видны только самому яблоку и его клонам. Коты и цветочек не будут знать о существовании этих переменных, и никак не смогут их изменить.

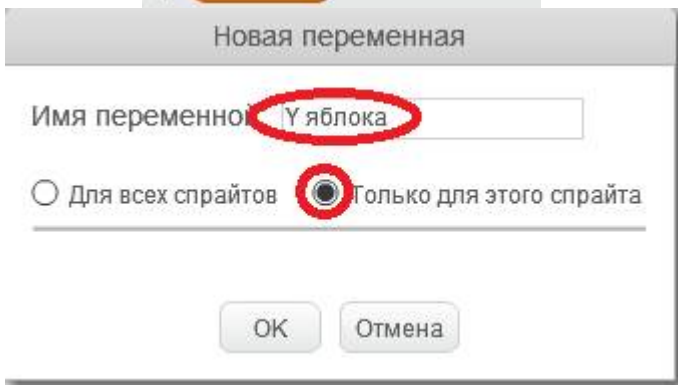



Чтобы создать локальную переменную, надо поставить точку «Только для этого спрайта».




Создайте вторую переменную.




Локальные переменные готовы. Можете зайти в любого кота, и убедиться, что у него эти переменные не отображаются. Их видит только Яблоко. Глобальные переменные «очки 1» и «очки 2» видны всем спрайтам.

Переменные готовы, можно делать программу для Яблока. Создайте первый скрипт, он начинает работать, когда получит сообщение «рассыпать яблоки». Скрипт задаёт случайные координаты X и Y, перемещает туда яблоко, и создаёт в этом месте его клон. Клон — это точная копия спрайта Яблоко.

когда я получу рассыпать яблоки
повторить 20
задать X яблока значение выдать случайное от -500 до 500
задать Y яблока значение выдать случайное от -300 до 300
перейти в x: X яблока y: Y яблока
создать клон себя самого
скрыть

Скрипты, управляющие клонами, всегда начинаются с блока «когда я начинаю как клон». Таких скриптов у яблока четыре штуки. Первый из них служит для организации скроллинга. При нажатии на стрелки клоны будут смещаться. Да-да, все клоны будут одновременно выполнять одно и то же действие, этот скрипт управляет сразу всеми клонами Яблока. Если Вы нажмёте стрелку вверх, то все клоны Яблока поедут вниз, Y изменится на минус три. Здесь нет ошибки, стрелка вверх - Y изменится на минус. Благодаря этому всеобщему движению яблок вниз, будет казаться, что кот идёт вверх! Коты на протяжении всей игры будут находиться на одном и том же месте экрана. Васька в левой части экрана, Рыжик в правой части экрана, но, несмотря на это, игроку будет казаться, что коты, как сумасшедшие, носятся по огороду, и собирают яблоки.

когда я начинаю как клон
перейти назад на 1 слоев
показать
всегда
если клавиша стрелка вверх нажата? то
изменить Y яблока на -3
если клавиша стрелка вниз нажата? то
изменить Y яблока на 3
если клавиша стрелка направо нажата? то
изменить X яблока на -3
если клавиша стрелка влево нажата? то
изменить X яблока на 3

Следующий скрипт позволяет яблокам, доплывшим до края экрана спрятаться, как будто они просто скрылись из вида за пределами экрана. Если этот скрипт не сделать, то при движении котов много яблок будет скапливаться у границы экрана. После создания игры попробуйте «выключить» этот скрипт, оторвав от него первый блок «когда я начинаю как клон». Размер экрана в Scratch, как Вы помните, от -240 до 240 по горизонтали, и от -180 до 180 по вертикали, однако в следующем скрипте числа на десять меньше. Это сделано для того, чтобы движения клонов были более плавными.

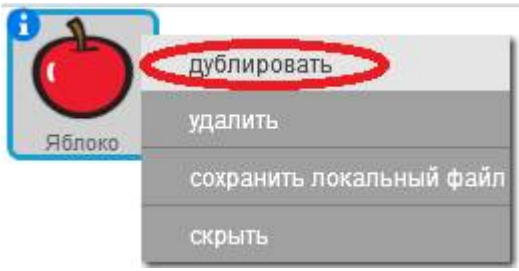
когда я начинаю как клон
всегда
если X яблока > 230 или X яблока < -230 или Y яблока < -170 или Y яблока > 170 то
скрыть
иначе
показать
перейти в x: X яблока y: Y яблока

Следующие два скрипта считают количество собранных яблок. Этот скрипт считает, сколько яблок собрал Васька. Каждый раз при касании очередного клона Яблока Васькой, переменная «очки 1» увеличится на единицу, клон скроется, и, через десятую долю секунды, удалится.

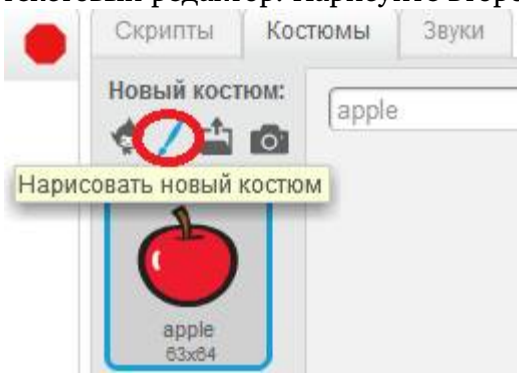
Этот скрипт считает, сколько яблок собрал Рыжик. Работает он, так же, как и предыдущий.

Шестой скрипт Яблока, при получении сообщения о том, что теперь настала очередь Рыжика собирать яблоки, удаляет клоны яблока, которые не успел собрать Васька. Чтобы лучше понять, зачем здесь этот блок, попробуйте удалить его, и запустить игру без него. Увидите, что будет.

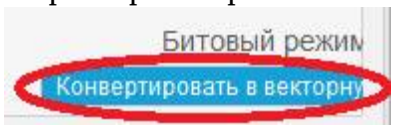
Скрипты яблока готовы. Можете немного поиграть. Коты ходят, и собирают яблоки, а когда яблоки кончаются, то коты гуляют по пустому полю. Когда поле пустое, яблок нет, и ничего не движется, то эффект скроллинга пропадает. Несмотря на то, что Вы нажимаете стрелки, кажется, что кот стоит на месте. Для того, чтобы эффект скроллинга никогда не пропадал, надо чтобы на поле были объекты, которые кот собирать не будет. Перемещение этих объектов будет делать движение по полю реалистичным даже при отсутствии яблок. Этим дополнительным предметом сделаем цветочек. В результате, кот будет гулять по саду, заросшему цветами, и собирать яблоки. Дублируйте яблоко, будем превращать его в цветочек.




Зайдите в текстовый редактор. Нарисуйте второй костюм




Перейдите в векторный режим рисования.




Увеличьте масштаб.

Выберите инструмент «Карандаш».

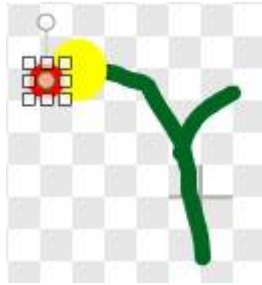
Зелёным цветом нарисуйте стебелёк.

Выберите инструмент «Эллипс».

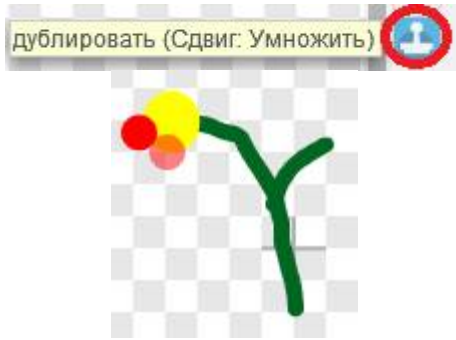
Нарисуйте жёлтый центр цветка — цветоложе.

Тем же инструментом «Эллипс» нарисуем красные лепесточки.

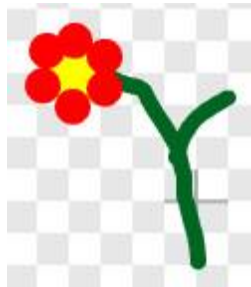
Сначала нарисуйте один лепесток.



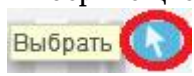
Пока лепесток выделен, нажмите на клавиатуре сочетание клавиш Ctrl-C, а потом Ctrl-V. Включится инструмент «дублировать».

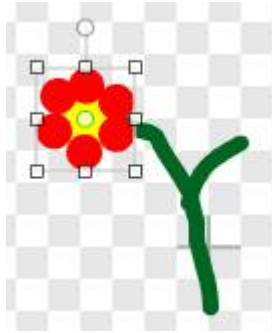



Удерживая нажатой клавишу Shift (в Scratch эта клавиша ошибочно переведена как Сдвиг), отпечатайте необходимое количество копий первого лепестка.

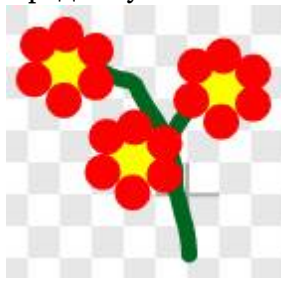


Инструментом выбора выберите цветок без стебля.

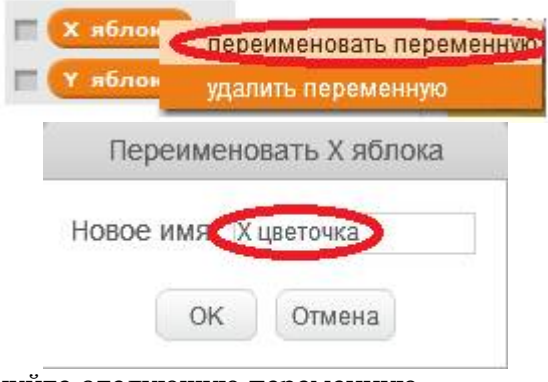




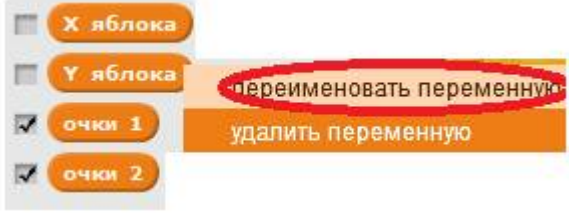
Нажмите на клавиатуре сочетание клавиш Ctrl-C, а потом Ctrl-V. Включится инструмент «дублировать». Скопируйте цветок на второй стебель, и в серединку.

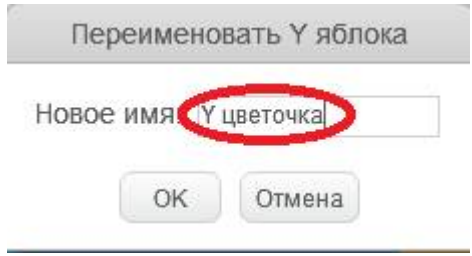


Цветочек готов. Сейчас у него такие же скрипты, как и у яблока, и такие же локальные переменные. Локальные переменные заново создавать не нужно, их нужно переименовать. Кликните правой кнопкой в переменную «X яблока», и выберите «переименовать переменную».

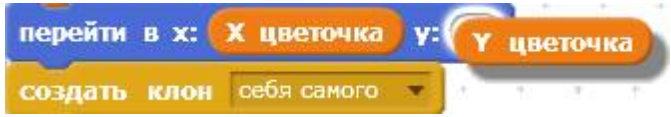



Переименуйте следующую переменную.

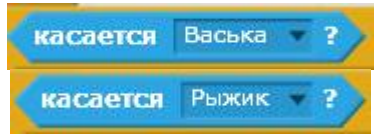





После того, как мы переименовали переменные, они повыпрыгивали из своих мест. Теперь, каждую переменную надо аккуратно положить на место.



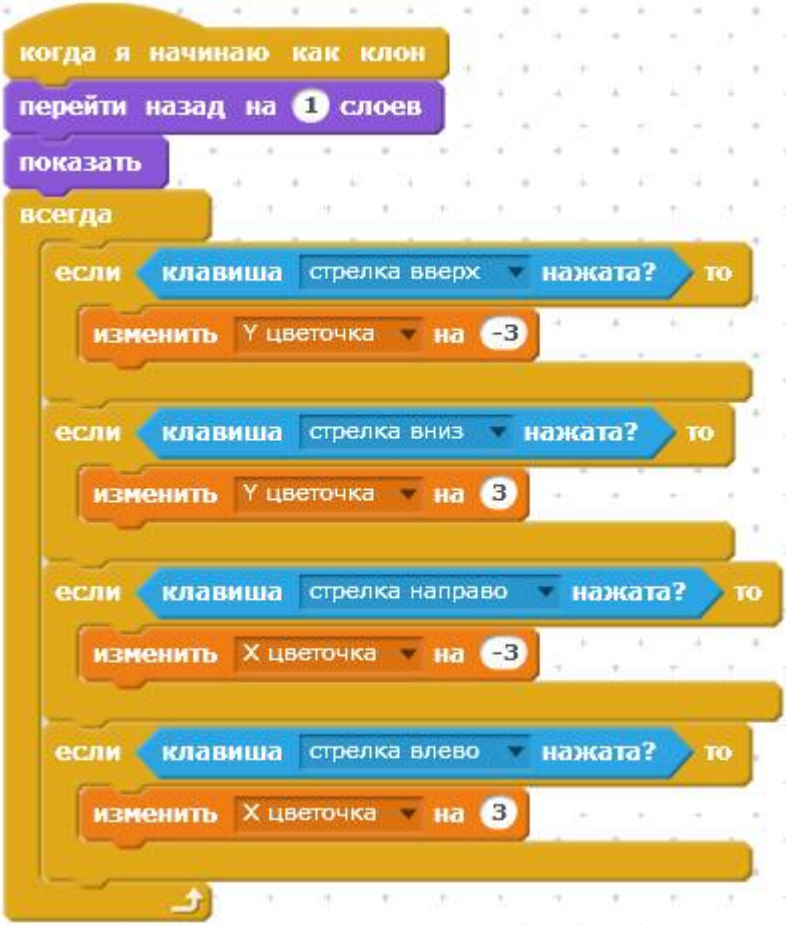

Скрипты цветочка надо немного изменить, чтобы коты на них не реагировали. Удалите у цветочка скрипты, отвечающие за взаимодействие с котами. Эти скрипты содержат блок «если касается Васька» и «если касается Рыжик».




Должны остаться только четыре скрипта. Скрипт, размещающий цветочки на поле.

когда я получу рассыпать яблоки
повторить 30
задать X цветочка значение выдать случайное от -500 до 500
задать Y цветочка значение выдать случайное от -300 до 300
перейти в x: X цветочка y: Y цветочка
создать клон myself
скрыть

Скрипт управления движением.


когда я начинаю как клон
перейти назад на 1 слоев
показать
всегда
если клавиша стрелка вверх нажата? то
изменить Y цветочка на -3
если клавиша стрелка вниз нажата? то
изменить Y цветочка на 3
если клавиша стрелка направо нажата? то
изменить X цветочка на -3
если клавиша стрелка влево нажата? то
изменить X цветочка на 3

Скрипт ухода за границу экрана.

когда я начинаю как клон
всегда
если X цветочка > 230 или X цветочка < -230 или Y цветочка < -170 или Y цветочка > 170 то
скрыть
иначе
показать
перейти в x: X цветочка y: Y цветочка

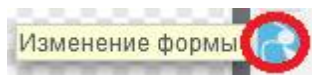
И скрипт, удаляющий клонов.

Игра почти готова. Мне не нравится, что коты у нас получились какими-то одинаковыми. Разве бывают в одном доме одинаковые коты? Давайте перекрасим Ваську в серый цвет, и изменим ему ушки. Он будет шотландской вислоухой породы.

Выберите Ваську и зайдите в графический редактор. Выберите серый цвет, и инструмент «Окрасить форму».

Окрасьте Ваську.

Осталось сделать ушки-вислоушки. Выберите инструмент «Изменение формы».




Кликните на ухо.

О ужас! Кот покрылся волдырями!

Да ладно, это не волдыри, а опорные точки. Если потянуть за одну из этих точек, то мордочка Васьки изменит свою форму. Давайте, изменим форму ушек, придадим котику немного шотландского шарма.

Вот теперь всё готово. Можно поиграть.

Играть в игру лучше всего вдвоём. Вы играйте за Ваську, а Ваш друг пусть играет за Рыжика. Кто из вас наберёт больше очков?

Вопросы по игре «Собираем яблочки».

1. Как ускорить движение котов?
2. Как увеличить количество яблок?
3. Как увеличить количество цветов?

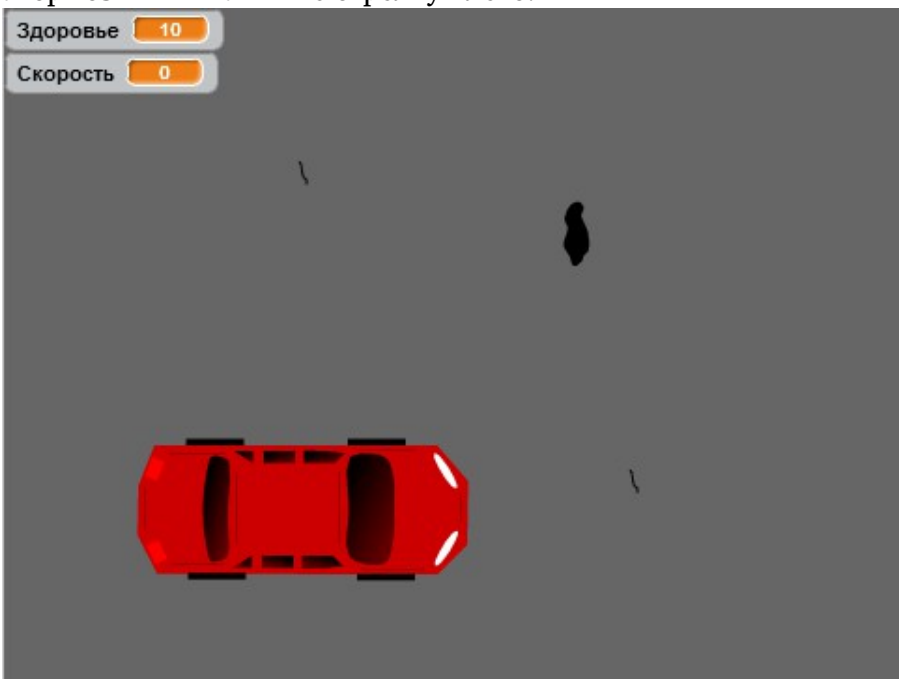
Задания к игре «Собираем яблочки».

1. Сделайте так, чтобы в саду росли цветы двух видов.
2. Добавьте в игру испорченные яблоки. При касании этого яблока очки должны не увеличиться, а уменьшиться на 1.
3. Добавьте в игру третьего Кота и переменную «очки 3», можно будет играть втроём.

# ГЛАВА 4. СТРИТРЕЙСИНГ.

Решил Кот поучаствовать в стритрейсинге. Купил красную машину, заправил, добавил закиси азота для скорости. Длина трассы пять километров. За какое время Кот сможет её преодолеть?

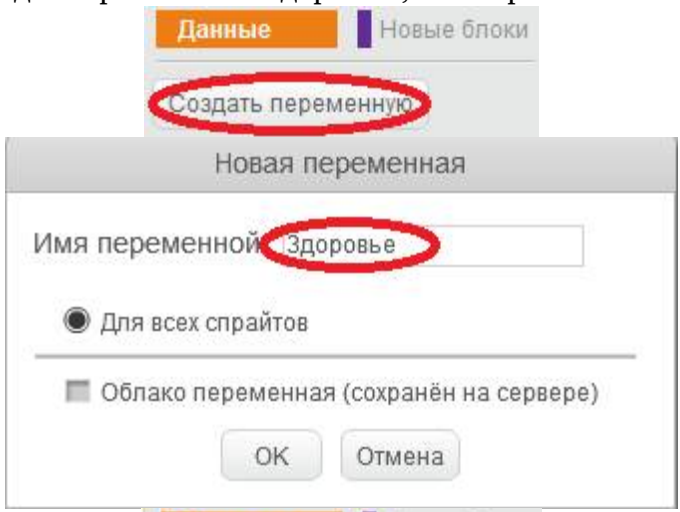
Правила игры такие: машина едет только по прямой, поворотов нет. На дороге попадаются опасные ямы и трещины. При попадании в яму, «Здоровье» игрока уменьшается на единицу. Старайтесь в них не попадать! Управление автомобилем осуществляется с помощью стрелок вверх и вниз на клавиатуре. Скорость увеличивается нажатием стрелки вправо. Чтобы затормозить — нажмите стрелку влево.

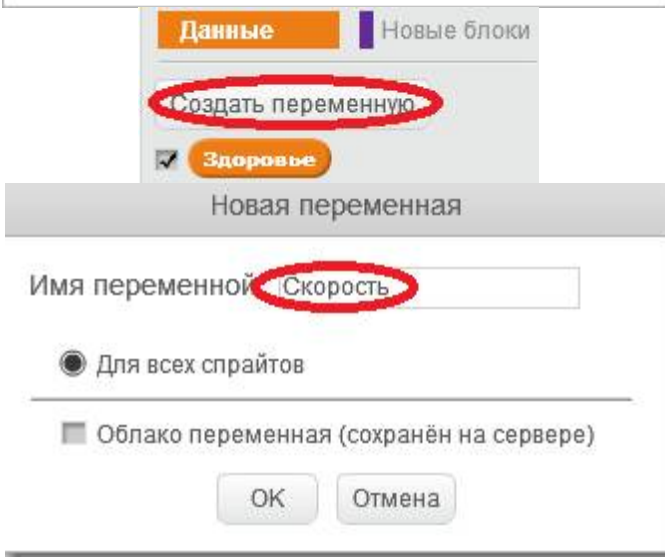



Всего в этой игре у нас будет четыре спрайта.

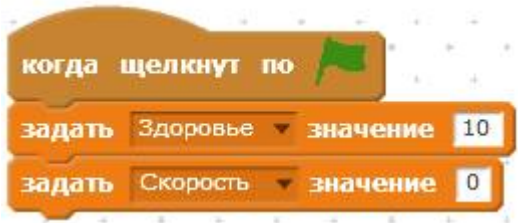
Сначала нарисуйте фон. Просто залейте сцену серым асфальтовым цветом.

Сразу сделайте маленькую программу для сцены. В этой программе просто задаются начальные значения переменным. Создайте две переменные: «Здоровье», и «Скорость».

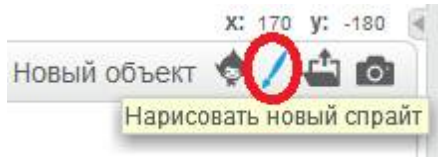


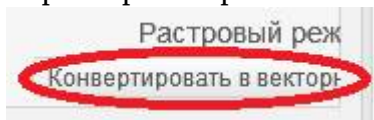

Сделайте вот такой скрипт для сцены.



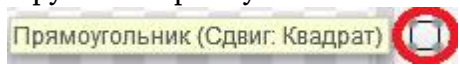

Теперь создадим спрайты. Сначала нарисуем Машину, потом Яму, Трещину и Финиш. Когда все спрайты будут готовы, запрограммируем их. Нажмите кнопку «Нарисовать новый спрайт».




Перейдите в векторный режим рисования.

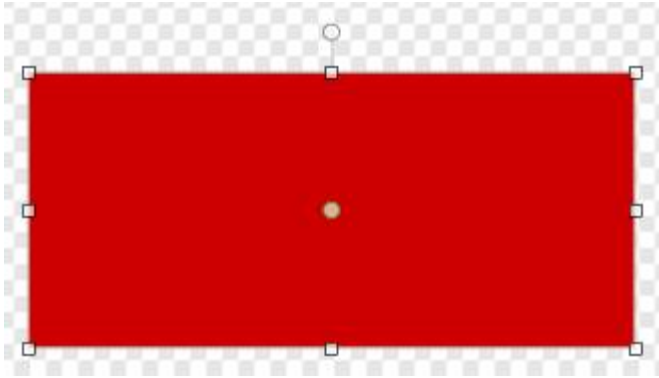



Выберите инструмент «Прямоугольник».




Выберите красный цвет.

Нарисуйте красный прямоугольник в центре.



Выберите инструмент «Изменение формы».

Станут видны опорные точки по углам прямоугольника. Кликните на границу прямоугольника, появится новая опорная точка. Переместите её.

Кликайте на границе прямоугольника, создавайте новые опорные точки, и придайте прямоугольнику такую форму.

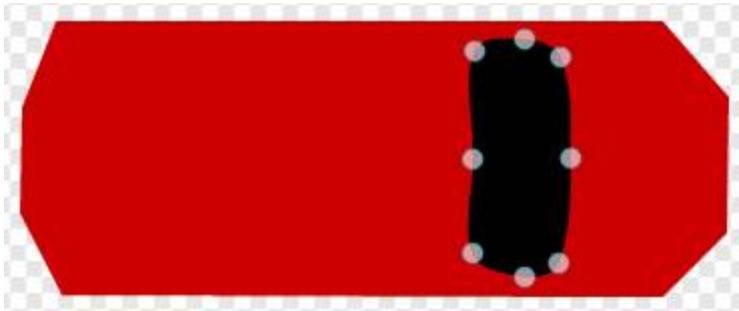
Теперь выберите инструмент «Эллипс».

Выберите чёрный цвет.

Нарисуйте чёрный овал. Из него мы сделаем лобовое стекло.

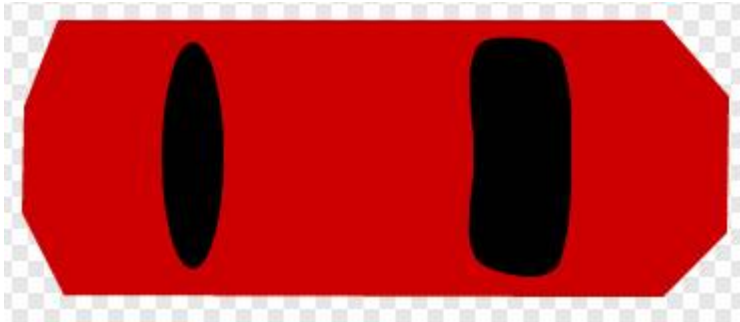
Снова выберите инструмент «Изменение формы».

Передвигая опорные точки, измените форму овала так, чтобы он стал похож на лобовое стекло.

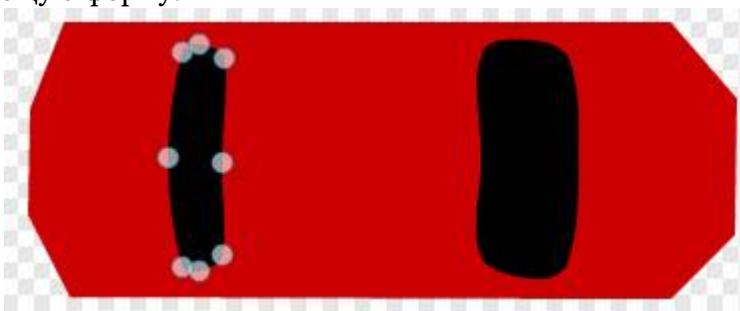


Снова выберите инструмент «Эллипс».

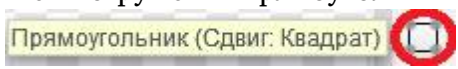
Нарисуйте тонкий овал на месте заднего стекла.



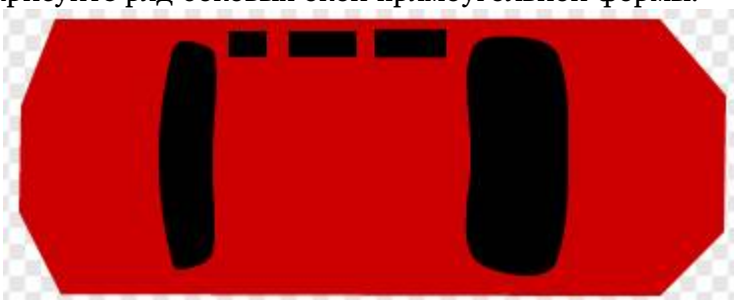
Инструментом «Изменение формы» придайте овалу следующую форму.



Снова выберите инструмент «Прямоугольник».



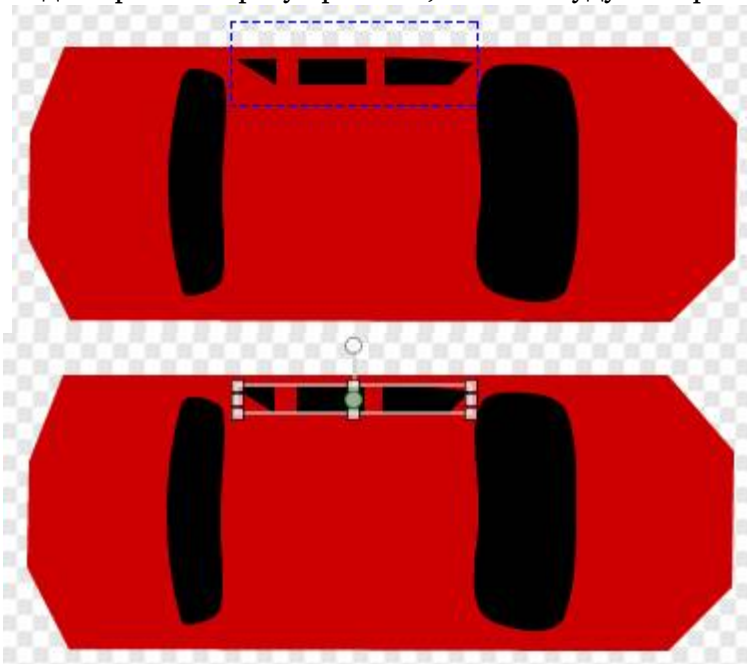

Нарисуйте ряд боковых окон прямоугольной формы.



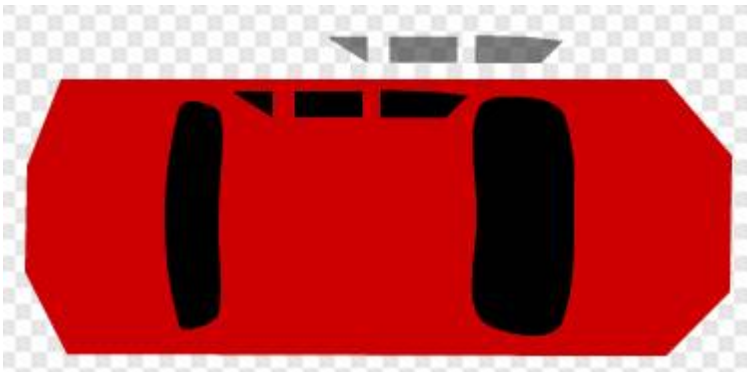
Снова выберите инструмент «Изменение формы» и измените форму боковых окон.

Инструментом выбора выберите боковые окна.

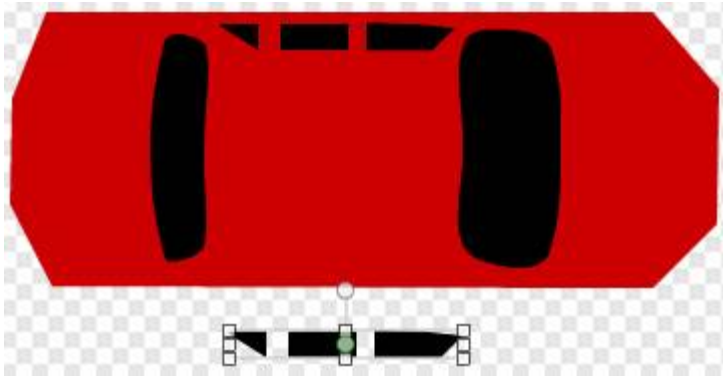
Обведите рамкой сразу три окна, все они будут выбраны.



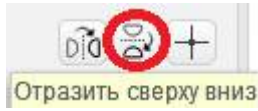
После того, как все три боковых окна выбраны, нажмите Ctrl-C, а после Ctrl-V. Автоматически включится инструмент «Дублирование», и к курсору мышки приклеится копия боковых окон.



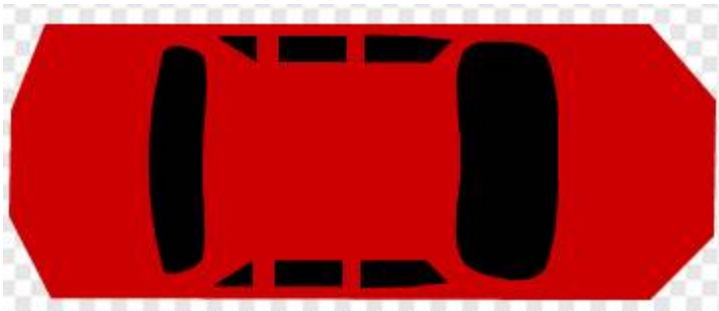
Отпечатайте где-нибудь копию боковых окон.



Пока копия боковых окон выделена, нажмите на инструмент зеркального отражения.

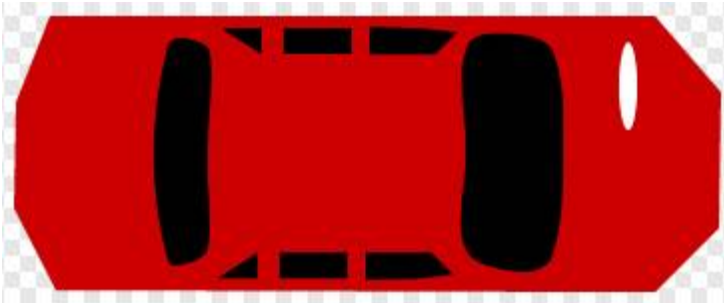



Окна перевернутся! Инструментом выбора подвиньте их на место. Благодаря инструментам дублирования и отражения, окна получились точно такие же.

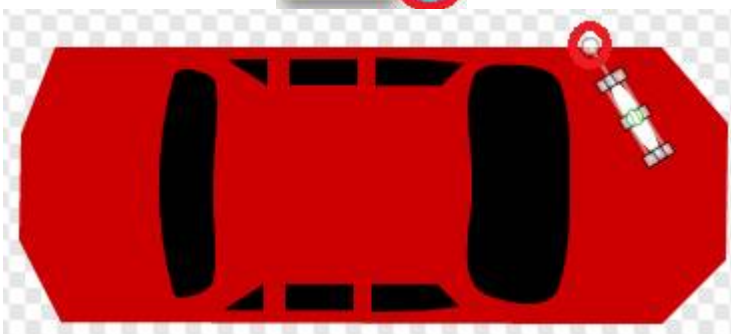


Сейчас можно нарисовать передние и задние фары. Передние фары нарисуем овальной формы, а задние — прямоугольной. Выберите инструмент «Эллипс».

Нарисуйте тоненький овальчик.

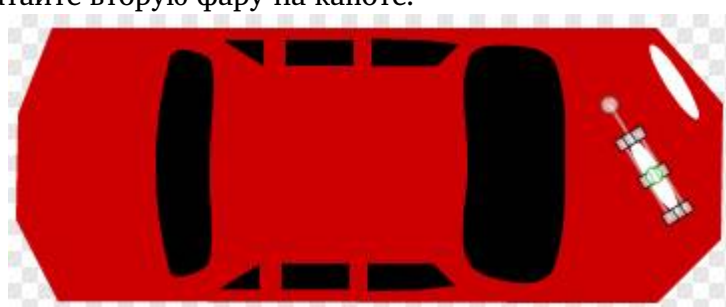


Выделите овальчик инструментом выбора, и поверните за точку.

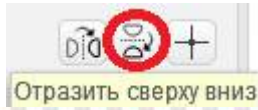


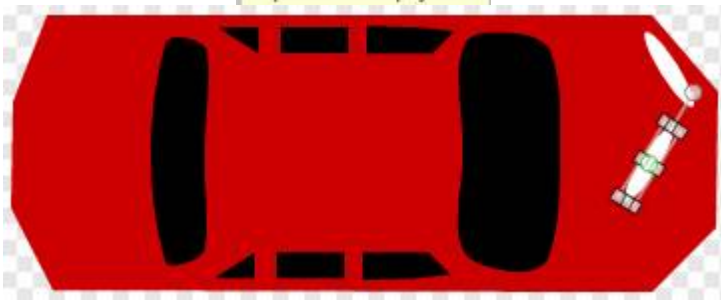
Подвиньте фару на место. Двигать мышкой не обязательно. Выделенный объект можно двигать, нажимая стрелочки на клавиатуре, так получается точнее.

Когда фара оказалась на месте, нажмите Ctrl-C и Ctrl-V. Отпечатайте вторую фару на капоте.

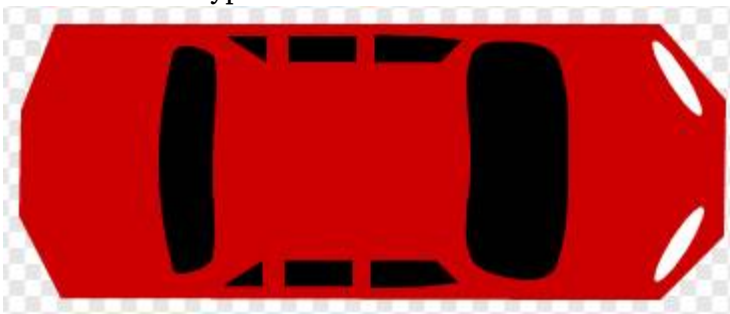


Пока фара выделена, нажмите на инструмент зеркального отражения.



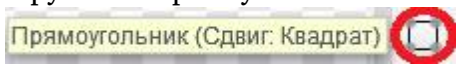



Подвиньте фару на место. Потренируйтесь использовать стрелочки на клавиатуре.

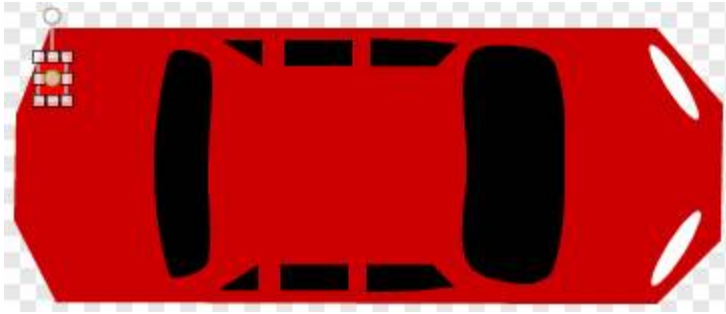


Задние фары нарисуем красными прямоугольничками.

Выберите инструмент «Прямоугольник».

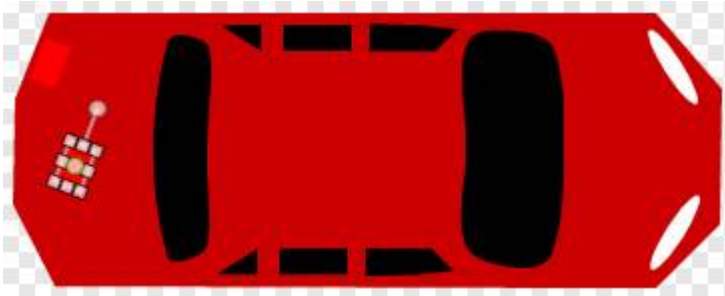



Выберите красный цвет.

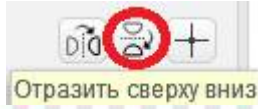


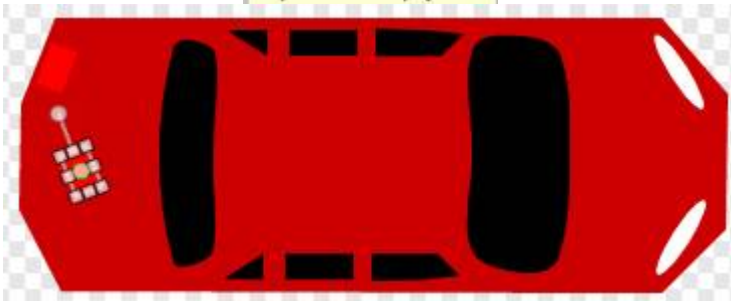
Поверните фару за кружочек.

Когда фара оказалась на месте, нажмите Ctrl-C и Ctrl-V. Отпечатайте вторую фару на капоте.



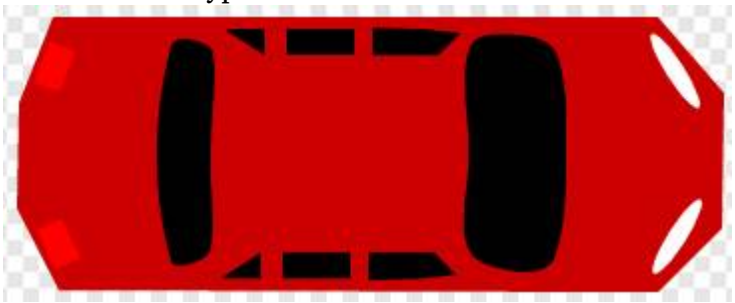
Пока фара выделена, нажмите на инструмент зеркального отражения.



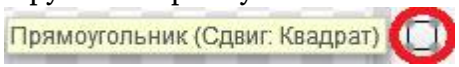
Подвиньте фару на место. Ещё раз попробуйте использовать

стрелочки на клавиатуре.



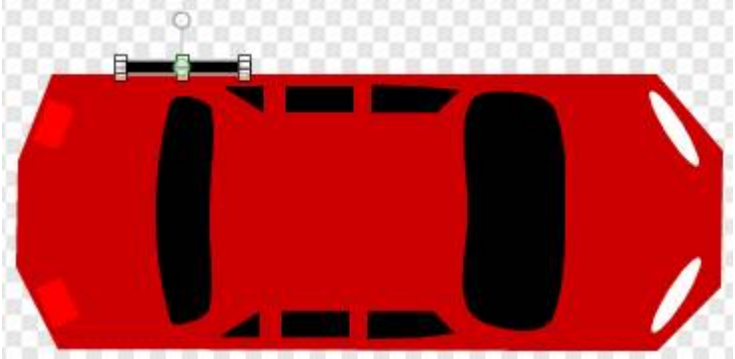
Автомобиль почти готов, осталось добавить колёса, и небольшие украшения — тюнинг. Колёса нарисуем тонкими чёрными прямоугольниками.

Выберите инструмент «Прямоугольник».

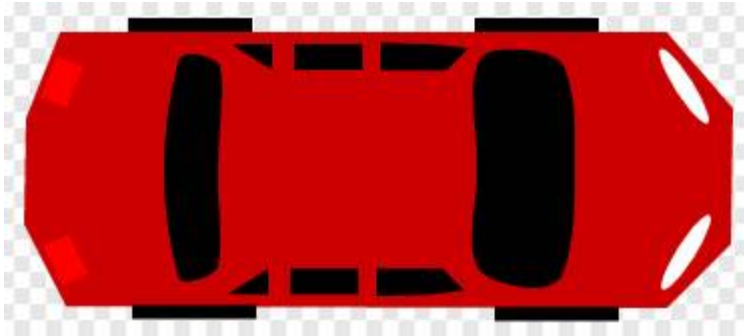



Выберите чёрный цвет.

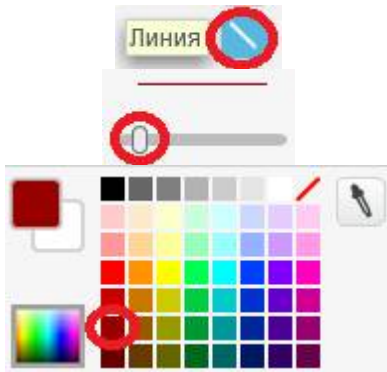
Нарисуйте одно колесо. Потом мы его скопируем три раза, чтобы все колёса были одинаковыми.



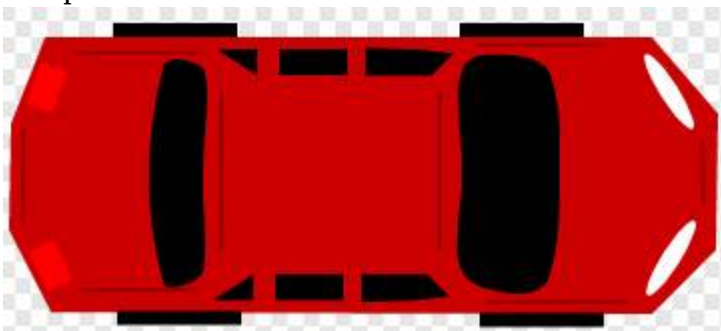
Пока колесо выделено, нажмите Ctrl-C и Ctrl-V. Отпечатайте второе колесо. Снова нажмите Ctrl-C и Ctrl-V, и отпечатайте третье колесо. Четвёртое колесо сделайте аналогично.



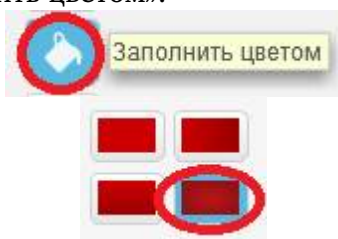
Остался небольшой тюнинг. Выберите линию тёмно-красного цвета.



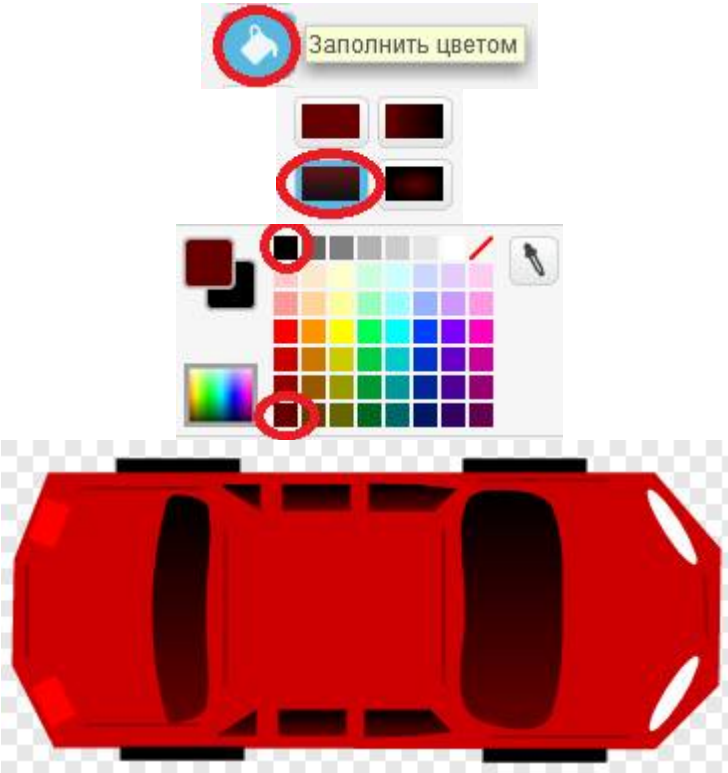

Рисуйте линии, удерживая клавишу Shift, так они будут получаться ровнее.



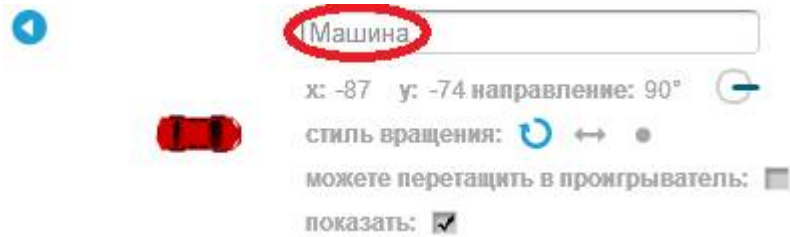
Получился классный автомобиль, но я предлагаю ещё немного его украсить — залить градиентной заливкой. Выберите инструмент «Заполнить цветом».



Машинка стала ещё краше. Последний штрих — залить градиентной заливкой стёкла.

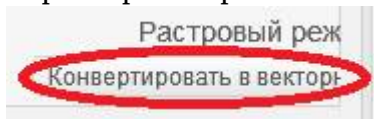



Здорово получилось! Какой кот откажется погонять на такой красивой машинке! Осталось только её переименовать.




Для того чтобы добавить в игру реализма, добавим на дорогу Ямы. Создайте новый спрайт.

Перейдите в векторный режим рисования.




Выберите инструмент «Эллипс».

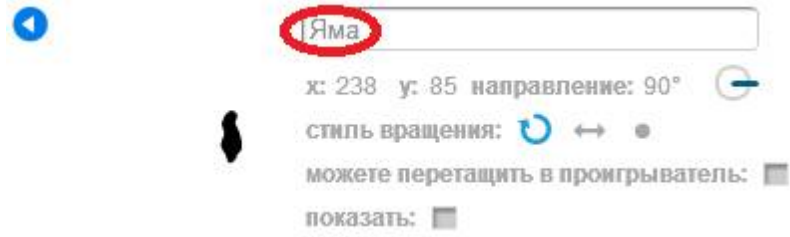
Выберите чёрный цвет.

Нарисуйте овальную Яму.

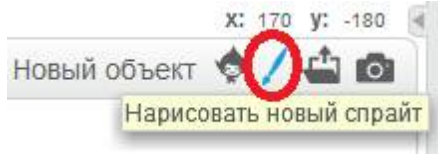
Выберите инструмент «Изменение формы».

Передвигая опорные точки, измените форму Ямы так, чтобы она стала похожа на обычную дорожную яму, которыми так богаты наши дороги.

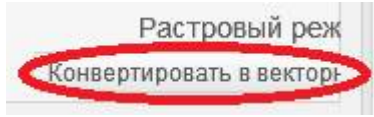
Яма готова. Осталось переименовать новый спрайт.




Яма готова. Осталось переименовать новый спрайт.




Перейдите в векторный режим рисования.

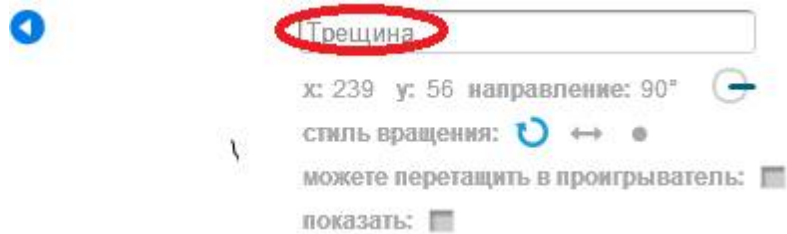



Перейдите в векторный режим рисования.

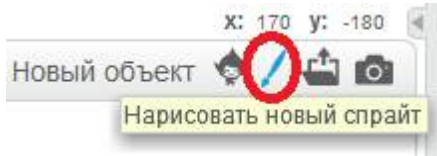
Выберите чёрный цвет.

Нарисуйте Трещину.

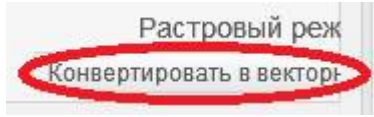
Переименуйте новый спрайт.



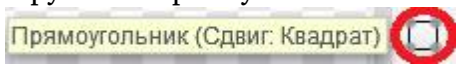

Четвёртый спрайт, это Финиш. Он выполнен в виде широкого оранжевого прямоугольника с надписью «Finish». К сожалению, русские буквы очень плохо отображаются в Scratch 2.0, поэтому писать всё будем по-английски. Создайте новый спрайт.




Перейдите в векторный режим рисования.




Выберите инструмент «Прямоугольник».




Выберите инструмент «Прямоугольник».

Возьмите оранжевый цвет.

Нарисуйте прямоугольник по высоте размером с экран.

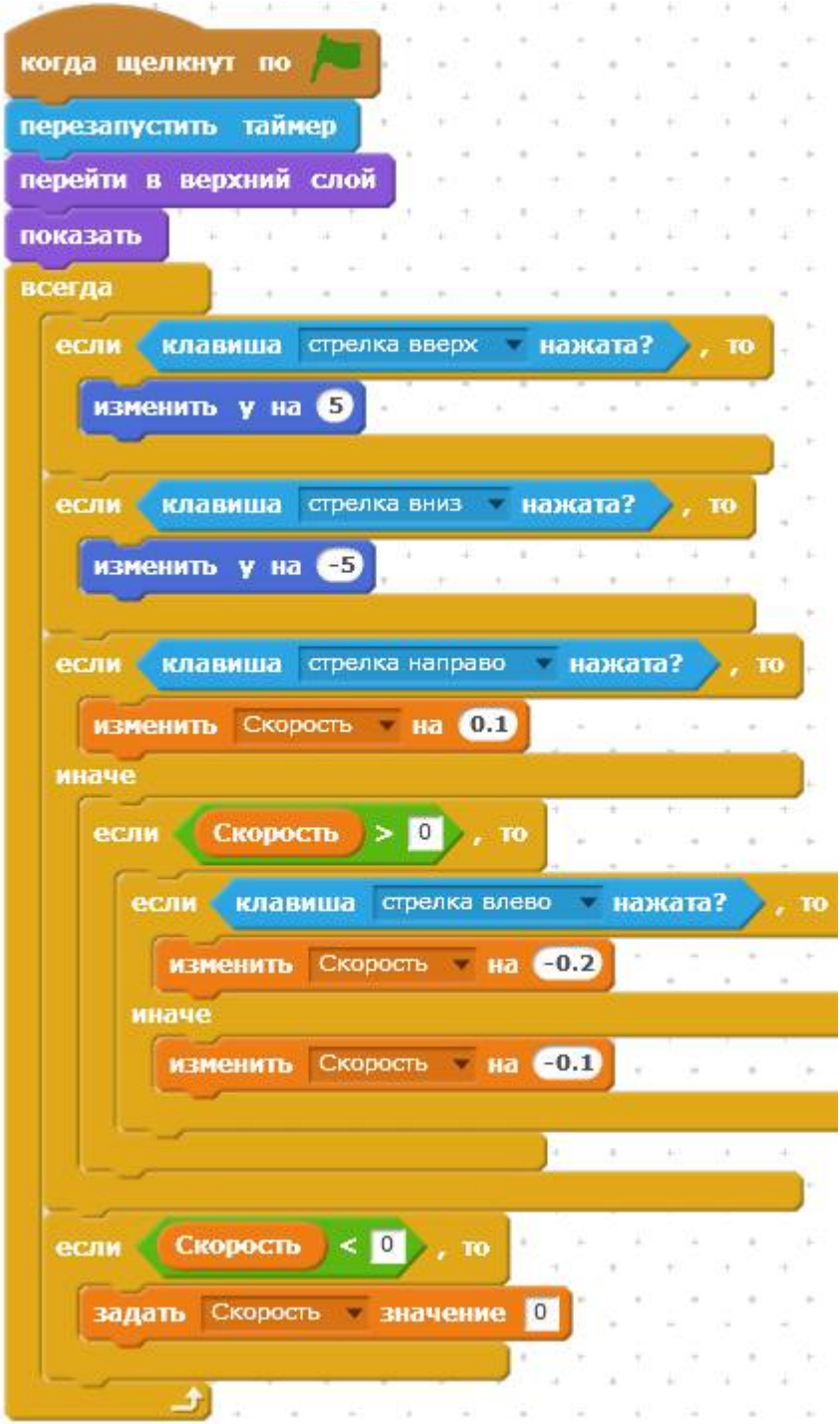
Выберите инструмент «Текст».

Напишите на прямоугольнике «Finish».

Все четыре спрайта готовы. Начинаем программировать!

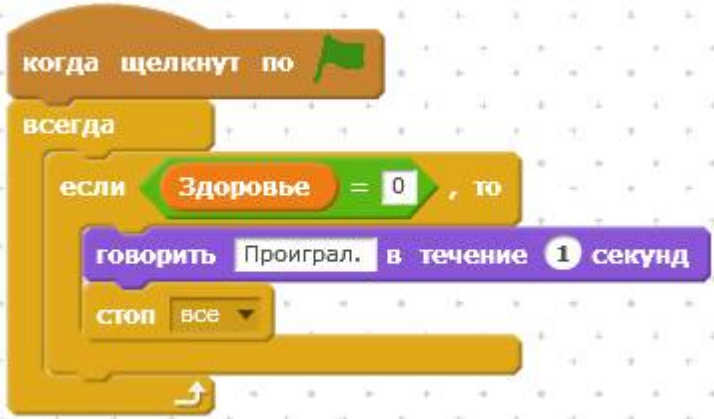
Сделаем программу для Машины. Она очень похожа на программу Яблока из игры «Собираем яблочки», в которой был вертикальный и горизонтальный скроллинг. Первый скрипт будет управлять вертикальным движением Машины и её ускорением. Он содержит четыре вложенных условия «Если». Первые два условия отвечают за перемещение Машины вверх и вниз. При нажатии на стрелку вверх, координата Y Машины будет увеличиваться, Машина будет перемещаться вверх. При нажатии на стрелку вниз всё будет наоборот, Машина будет опускаться. Следующее условие «Если — иначе» управляет скоростью автомобиля. Если стрелка вправо нажата, то скорость увеличивается на 0,1 километр в час каждую долю секунды. Это не очень быстро, такая же скорость разгона у моего Рено Логан.

Если стрелка вправо не нажата, то выполняется условие «иначе», в котором есть вложенное условие с проверкой нажатия стрелочки влево. Если стрелка влево нажата, то скорость автомобиля уменьшается на 0,2 километра в час каждую долю секунды. Машина тормозит в два раза быстрее, чем разгоняется. В том случае, если не нажата ни одна из стрелок управления скоростью, Машина едет плавно замедляясь — катится по инерции. Четвёртое условие не даёт опуститься скорости меньше ноля. Если значение переменной скорость станет меньше ноля, то оно тут же вернётся к нулевому значению. Это сделано для того, чтобы Машина не уехала обратно на старт, ведь скорость меньше ноля означает движение Машины в левую сторону, обратно на старт!

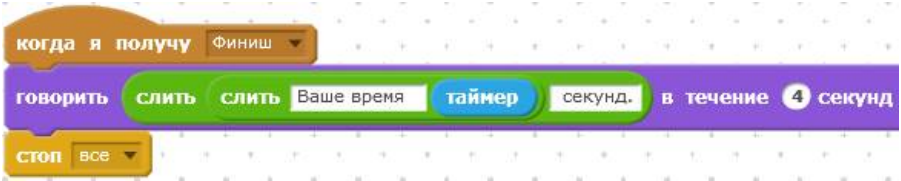

когда щелкнут по
перезапустить таймер
перейти в верхний слой
показать
всегда
если клавиша стрелка вверх нажата? , то
изменить y на 5
если клавиша стрелка вниз нажата? , то
изменить y на -5
если клавиша стрелка направо нажата? , то
изменить Скорость на 0.1
иначе
если Скорость > 0 , то
если клавиша стрелка влево нажата? , то
изменить Скорость на -0.2
иначе
изменить Скорость на -0.1
если Скорость < 0 , то
задать Скорость значение 0

Следующий скрипт определяет, не проиграл ли ты. Если ты проехал автомобилем по десяти ямам, то значение переменной «Здоровье» станет равно нулю. Это означает, что от Машины остались «рожки да ножки». Это конец игры.

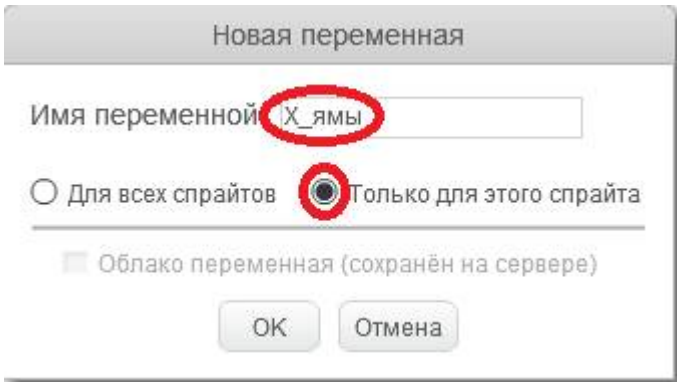
Машина скажет, что ты проиграл, и остановит выполнение программы.




Третий скрипт Машины объявляет время прохождения игры в случае победы.



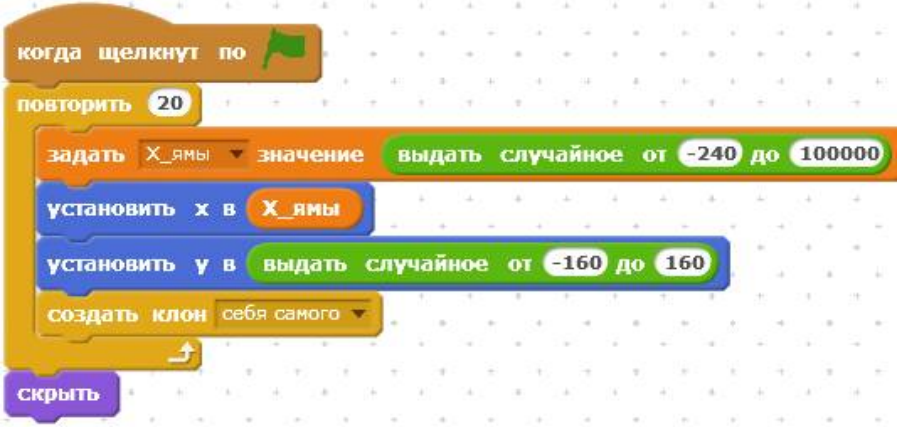

Перед тем, как приступать с программирования Ямы, создайте локальную переменную «X_ямы». Эта переменная будет видна только Яме и её клонам.



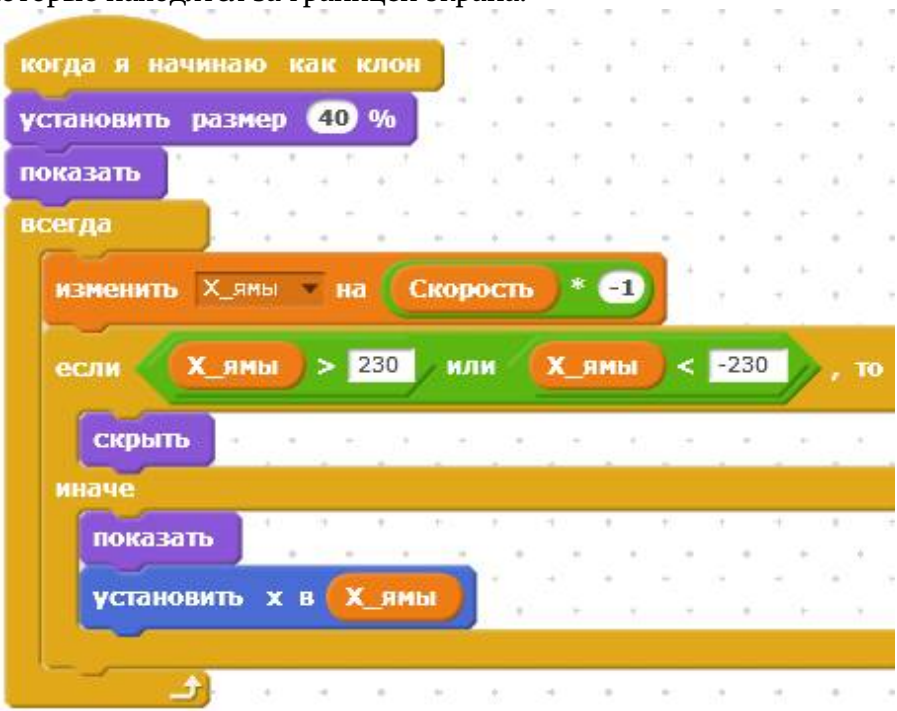

Как я уже говорил, программа Ямы очень похожа на программу Яблока из первой игры. Принципиальное отличие в том, что скроллинг в этой игре работает только по X, а значение Y изменяется только в пределах экрана, от -160 до 160. Скрипт работает очень просто. Он создаёт двадцать клонов себя самого, то есть двадцать ям на дороге, со случайной координатой X, и со случайной координатой Y. Значение 100000 означает длину дороги.

После того, как спрайт Ямы создаст двадцать своих клонов,

он скроется из вида.


когда щелкнут по
повторить 20
задать X_ямы значение выдать случайное от -240 до 100000
установить x в X_ямы
установить y в выдать случайное от -160 до 160
создать клон себя самого
скрыть

В следующем скрипте Ямы запрограммирован скроллинг и ускоренное движение автомобиля. Координата X Ямы изменяется не просто на число, как в игре про яблочки, а на переменную величину. В начале разгона, скорость мала, и координата X изменяется не сильно, Машина едет медленно. По мере увеличения скорости, координата X изменяется всё сильнее и сильнее. Машина летит вперёд, объехать Ямы становится труднее. Условие «Если», как и в прошлой игре скрывает дальние Ямы, которые находятся за границей экрана.

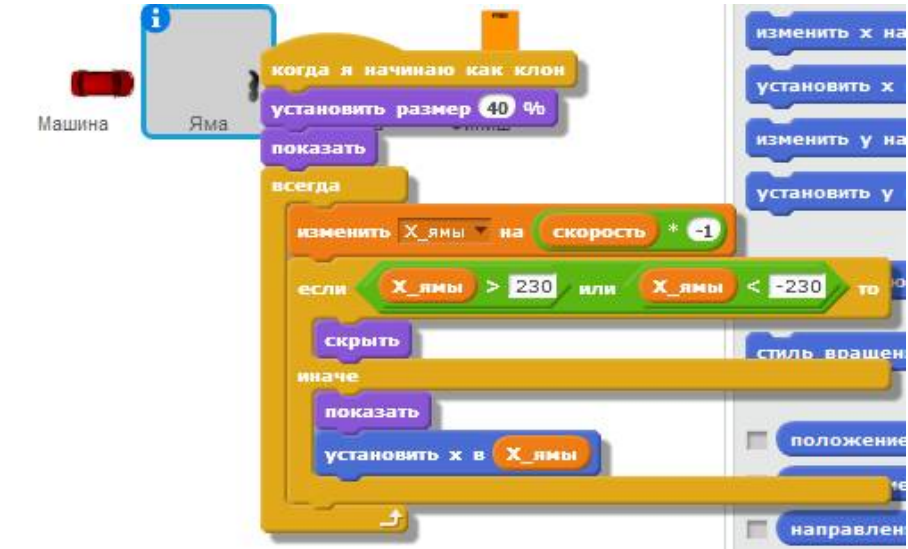



Следующий скрипт срабатывает в случае касания Машины и Ямы. Переменная «Здоровье» уменьшается на единицу, после

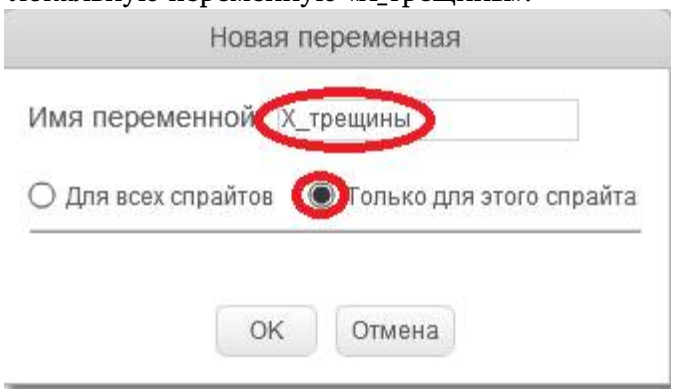
этого скрипт ждёт, пока Машина не перестанет касаться Ямы, и продолжает выполнение. Если блок «ждать до» выкинуть, то за время проезда Машины по Яме переменная «Здоровье» может измениться несколько раз.

Программа трещины очень похожа на программу Ямы. Чтобы облегчить создание программы, можно просто перетащить готовые скрипты Ямы на Трещину. Тащите готовый скрипт на Трещину до тех пор, пока курсор мышки не окажется точно над спрайтом Трещины. В отличие от Scratch 1.4, в Scratch 2.0, спрайт, в который копируется скрипт, не подсвечивается. Поэтому, при копировании скриптов, внимательно следите за положением курсора.

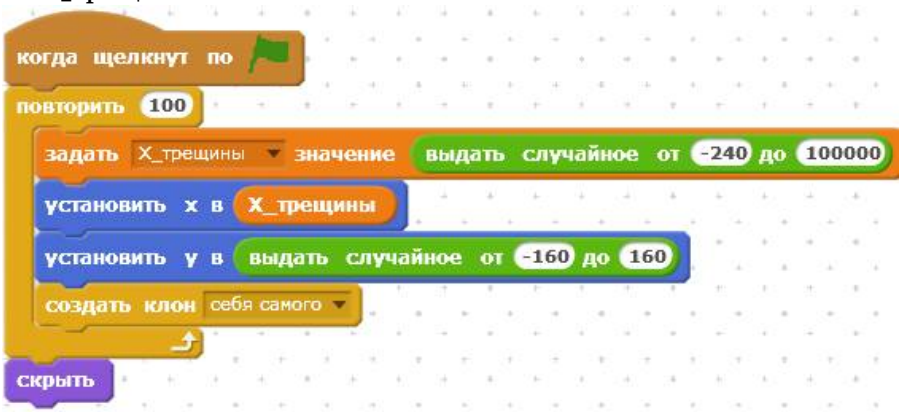
Теперь скопируйте второй скрипт Ямы. Отпускайте скрипт, когда курсор над спрайтом Трещины.




Теперь, когда скрипты скопированы, остаётся только заменить локальную переменную «X_ямы» на «X_трещины». Создайте локальную переменную «X_трещины».

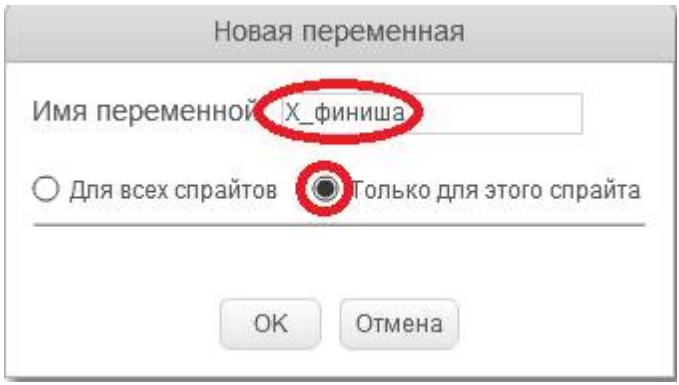



Замените в скопированном скрипте все переменные «X_ямы» на «X_трещины».


когда щелкнут по
повторить 100
задать X_трещины значение выдать случайное от -240 до 100000
установить x в X_трещины
установить y в выдать случайное от -160 до 160
создать клон себя самого
скрыть

Во втором скрипте Трещины сделайте аналогичную замену.

Осталось сделать только программу для Финиша. Сначала создайте локальную переменную «X_финиша». Как Вы догадываетесь, она будет содержать координату X спрайта Финиш.




Программа Финиша будет состоять из одного скрипта. В начале игры Финиш располагается на конце дороги. Его координата X равна 100000. Финиш, так же как Яма и Трещина, постоянно уменьшает значение переменной «X_финиша» на «Скорость», тем самым приближаясь к Машине.

Игра готова. Запускайте и попробуйте установить рекорд.

Вопросы.

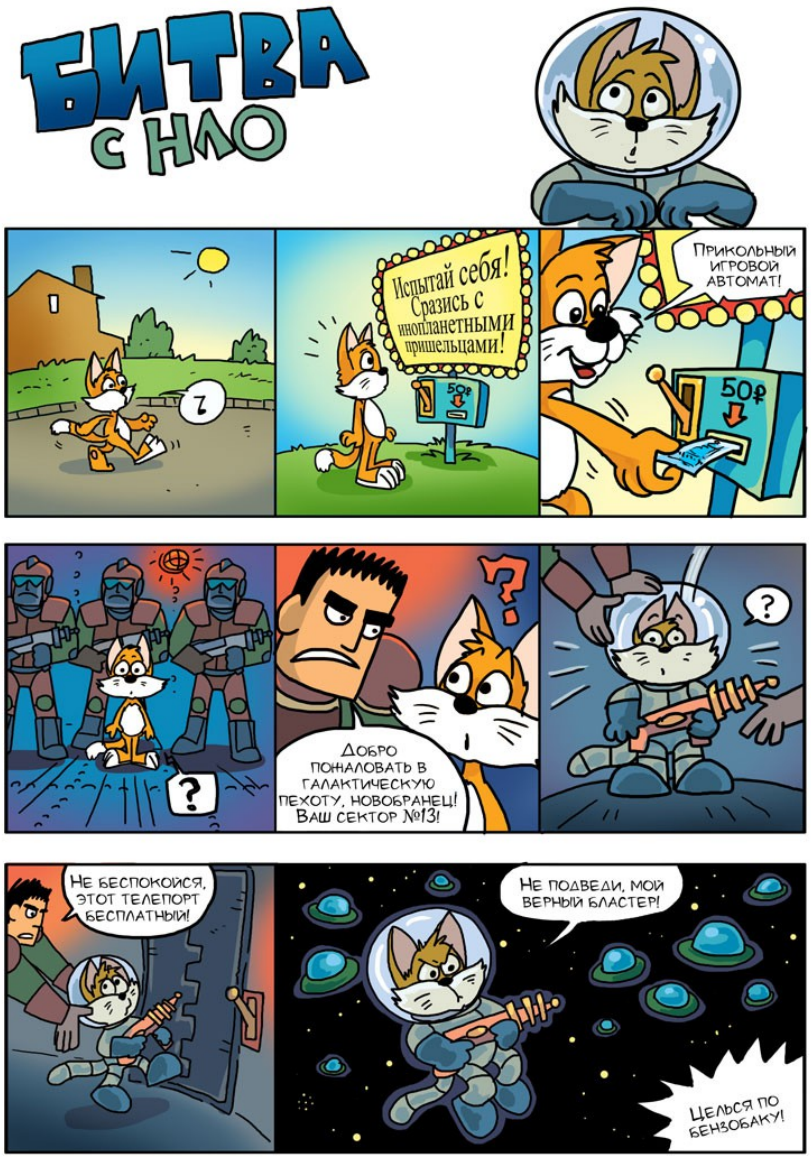
1. Как удлинить трассу?
2. Как сделать разгон автомобиля быстрее?
3. Как сделать торможение автомобиля медленнее?

Задание

1. Сделайте ограничение скорости в 90 км/ч, чтобы автомобиль не мог разогнаться быстрее.

2. Добавьте маленьких ямок, которые будут отнимать не 1, а 0,5 здоровья.

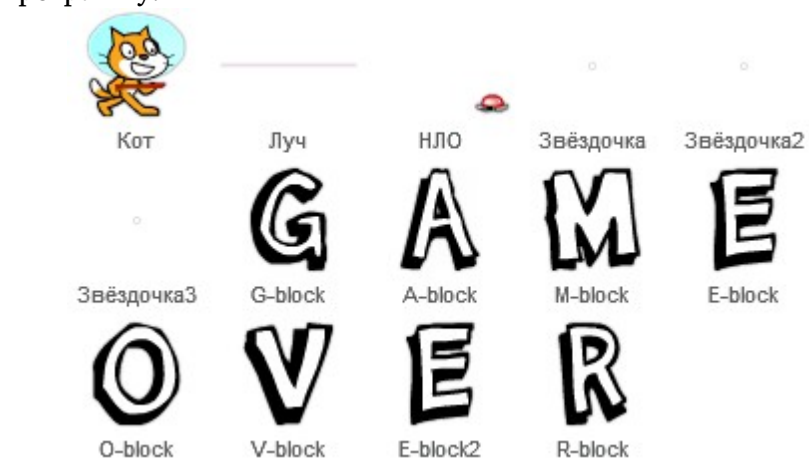
# ГЛАВА 5. КОСМИЧЕСКАЯ БИТВА.



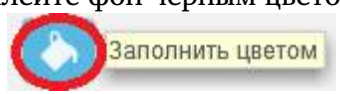

В XXXI веке Кот устроился работать космическим рейнджером. Его отправили охранять границу галактики от набегов злых Сыроманцев. Сыроманцы были очень опасны, они захватывали планеты, врывались в дома к беззащитным

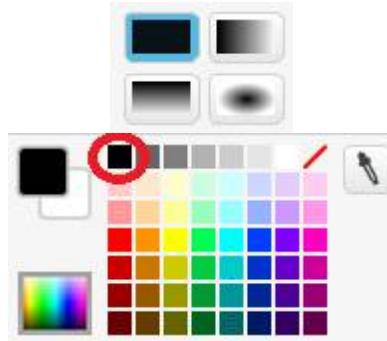
гражданам, и съедали всё из их холодильников! Злодеи должны быть остановлены!

В этой игре будет всего четырнадцать спрайтов. Кот, Луч бластера, вражеский НЛО, три звёздочки и восемь букв. На картинке Вы видите не один НЛО, а несколько. Это клоны одного спрайта, которые полностью похожи на него и имеют такую же программу.



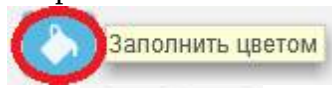

Создание игры, как обычно, начинаем с рисования сцены и спрайтов. Первым делом создадим два фона Сцене. Первый фон — чёрный космос. Залейте фон чёрным цветом.





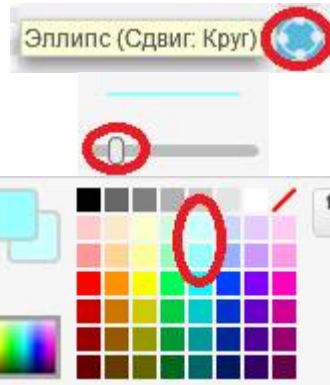
Теперь сделаем второй фон, он пригодится для заставки в конце игры. Нажмите кнопку «Нарисовать новый фон».

Залейте его чёрно-серым.

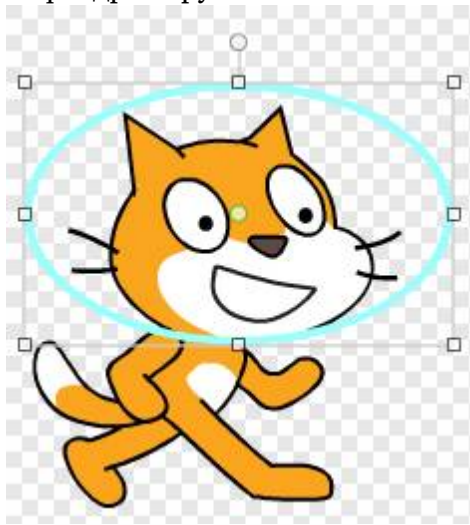


Получились два фона.

Теперь начинаем рисовать спрайты. Сначала оденем Кота в скафандр и дадим ему бластер. Выберите инструмент «Эллипс».

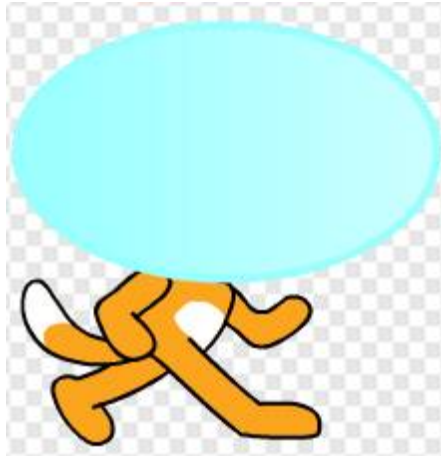



Нарисуйте скафандр вокруг головы Кота.



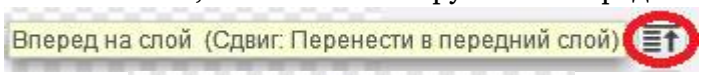
Выберите инструмент «Заполнить цветом».

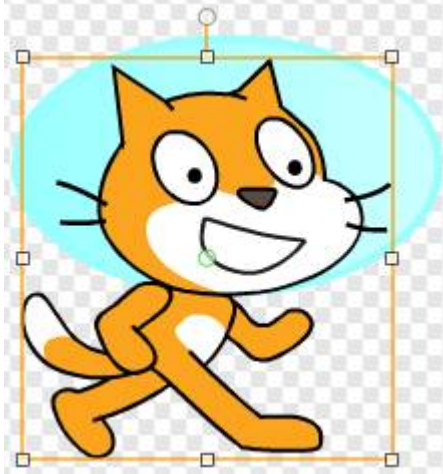
Получилось вот так.



Скорее! Кот не может дышать, надо срочно переместить Кота на слой выше, чтобы скафандр его не закрывал. Кликните инструмент выбора.

Кликните на Кота, а потом на инструмент «Вперёд на слой».

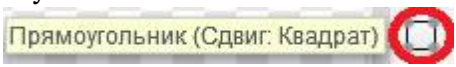


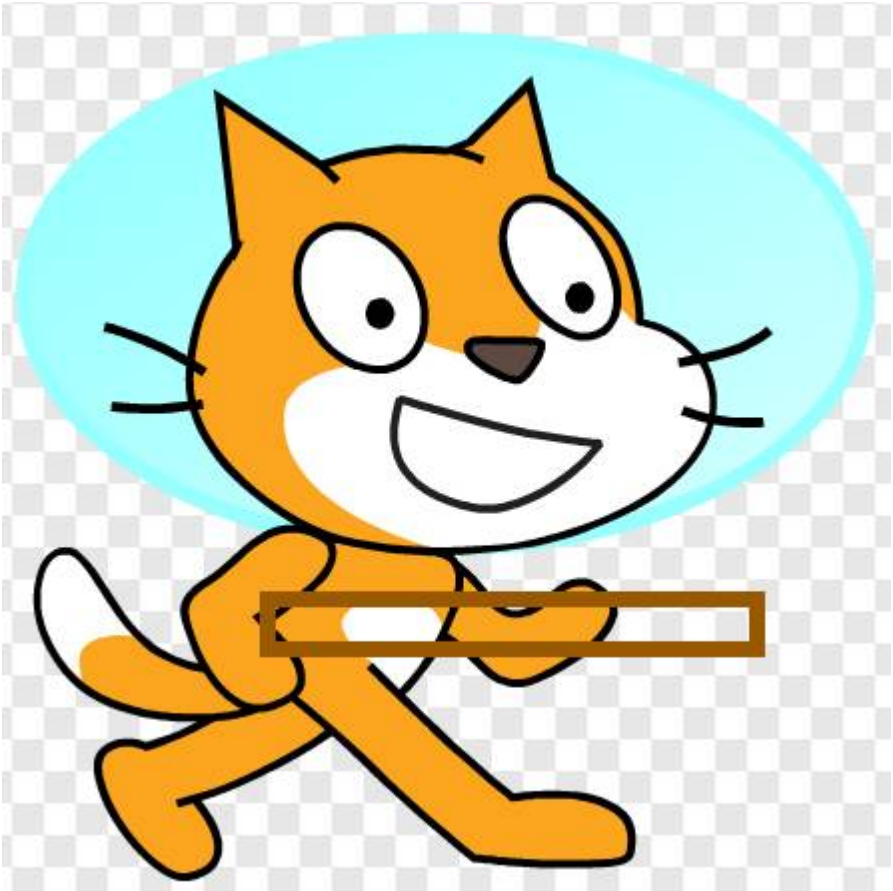



Кот спасён! В таком скафандре он может дышать в открытом космосе. Для защиты границы галактики надо вооружить Кота. Нарисуйте ему бластер. Например, такой.

Бластер рисуется с помощью прямоугольника, затем придайте ему необходимую форму, передвигая опорные точки. Увеличьте масштаб.

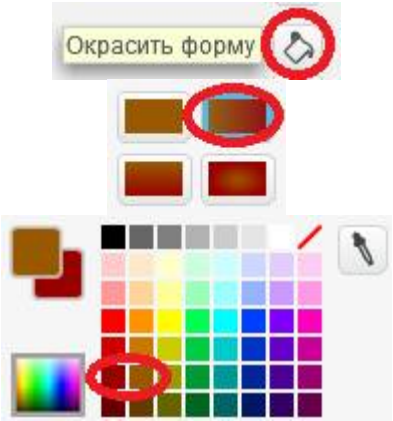
Сначала прямоугольник.



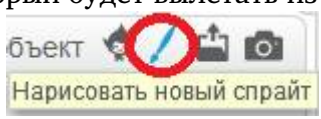


Потом кликните на «Изменение формы».

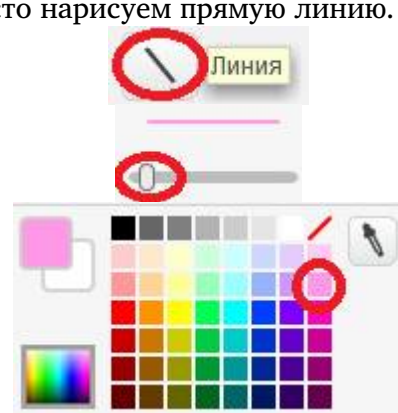
Залейте бластер градиентной заливкой.



Можете ещё немного украсить бластер, а я перехожу к рисованию Луча, который будет вылетать из бластера.

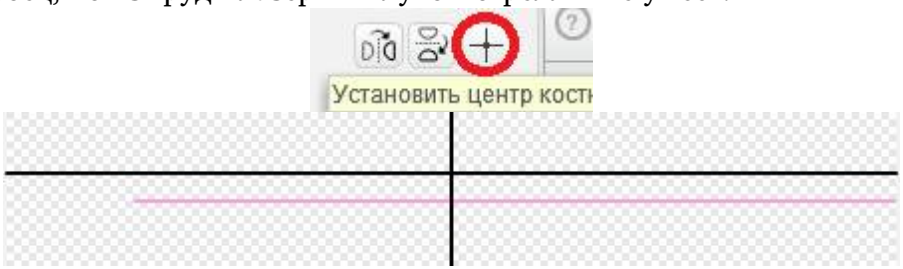



Здесь мы просто нарисуем прямую линию.

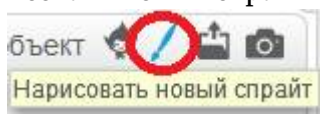



Нарисуйте прямую линию почти через весь экран, немного отступив слева. Чтобы линия получилась

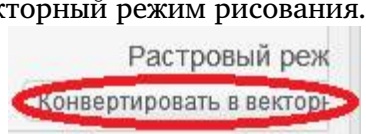
Для того, чтобы Луч вылетал точно из бластера, установите центр костюма немного повыше. Если этого не сделать, то Луч будет вылетать у Кота из груди. Наш Котик, конечно, крутой боец, но из груди лазерным лучом стрелять не умеет.



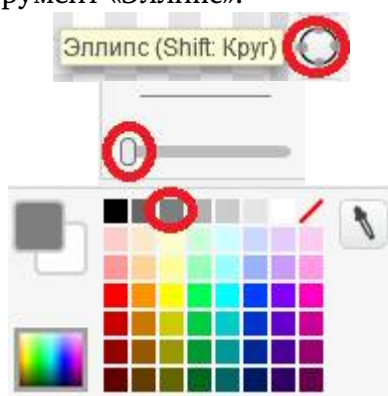

Теперь можно нарисовать инопланетных захватчиков. Нажмите кнопку «Нарисовать новый спрайт».



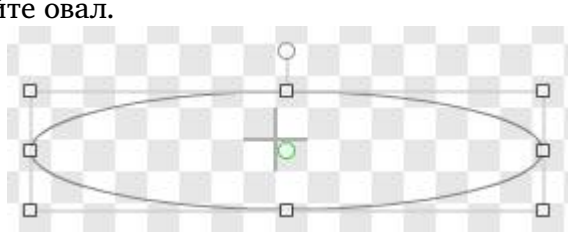

Перейдите в векторный режим рисования.




Выберите инструмент «Эллипс».

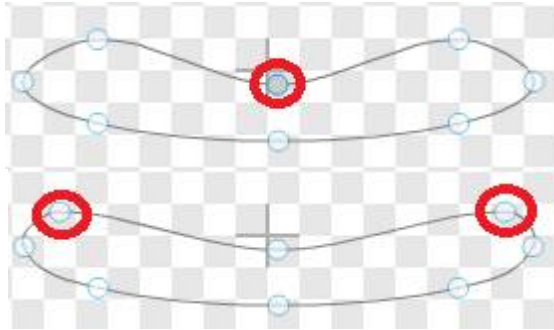



Нарисуйте овал.

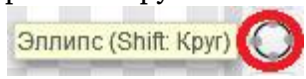


Выберите инструмент «Изменение формы».

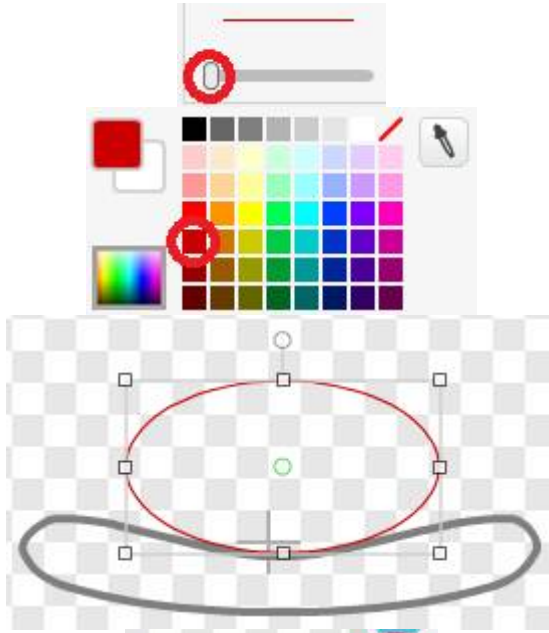
Перемещайте точки по очереди, чтобы получился корпус НЛО.



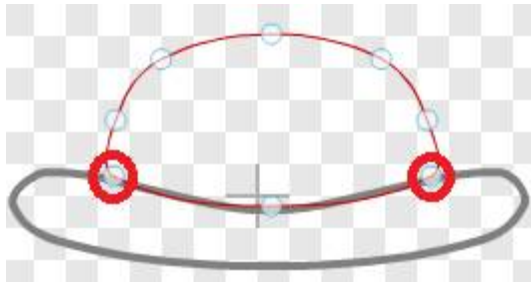
Теперь снова выберите инструмент «Эллипс».



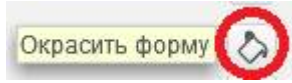

Нарисуйте заготовку стеклянного купола НЛО красным цветом.



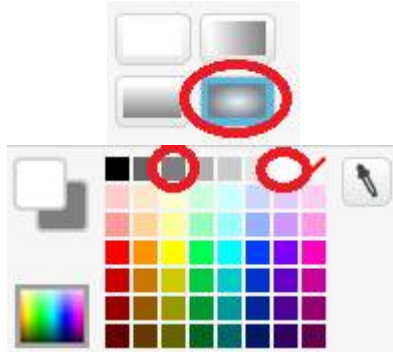
Поочерёдно перемещая опорные точки, придайте куполу правильную форму.

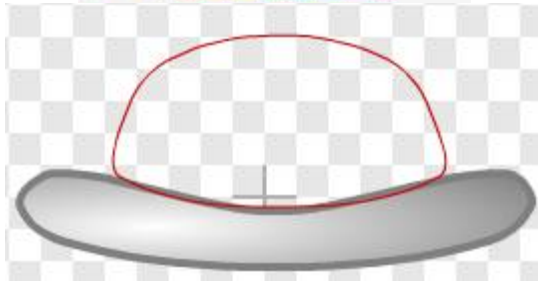


Выберите инструмент «Окрасить форму».

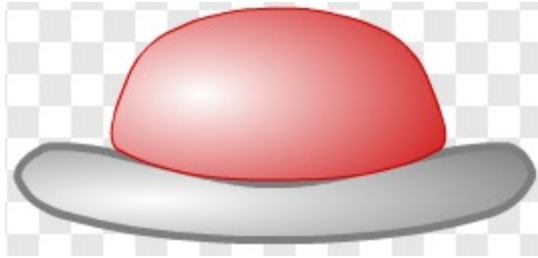



Окрасьте корпус серым цветом, а купол красным. Сначала корпус.

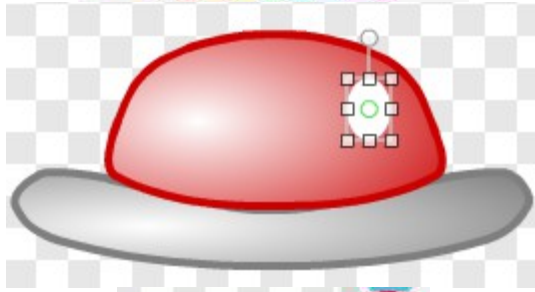


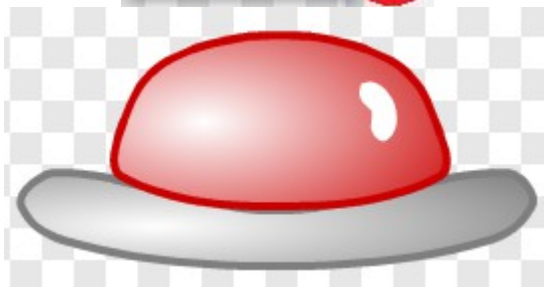


Затем купол.



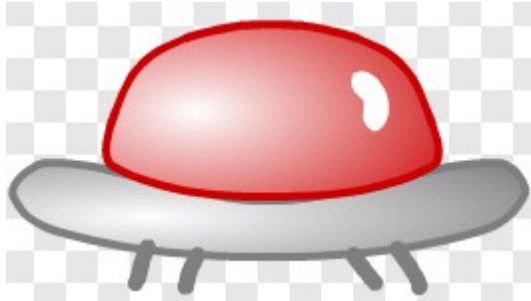
Теперь подрисуйте блик на куполе.





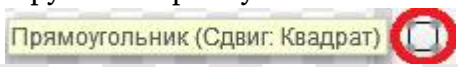
Теперь подрисуйте шасси. Возьмите «Карандаш».

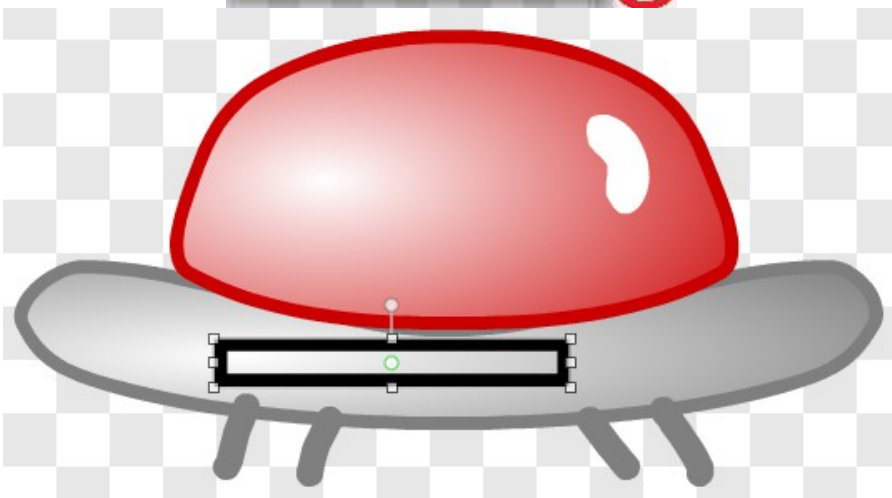
Подрисуйте шасси.



Теперь нарисуем ракетное вооружение. Увеличьте масштаб.

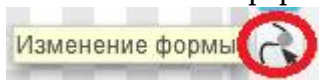
Выберите инструмент «Прямоугольник».



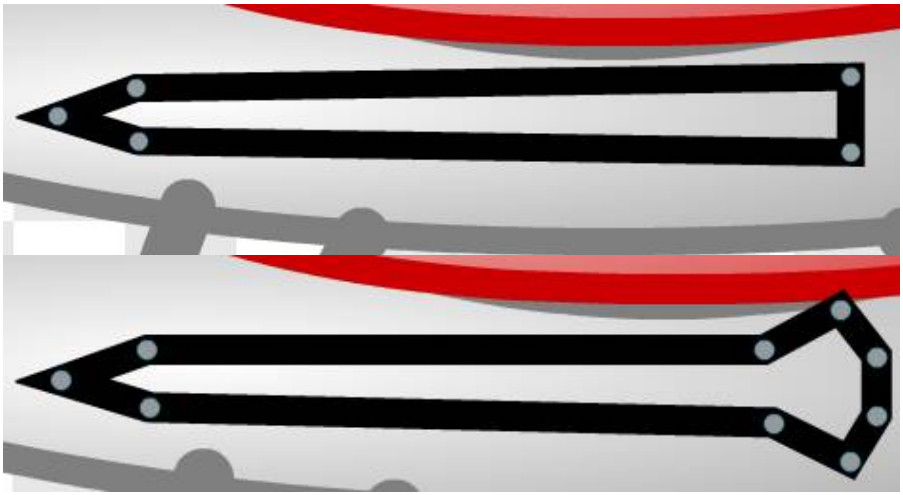



Ещё увеличьте масштаб.

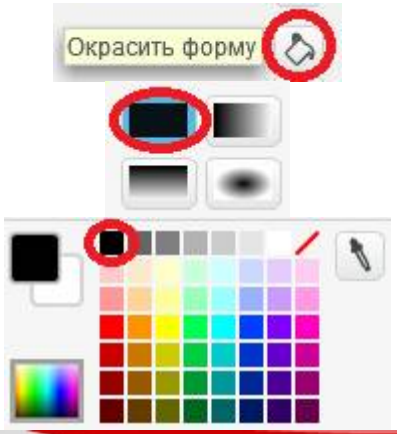
Выберите инструмент «Изменение формы».




Кликните в чёрную линию — появится новая опорная точка. Создавайте новые опорные точки и перемещайте их так, чтобы получилась ракета.

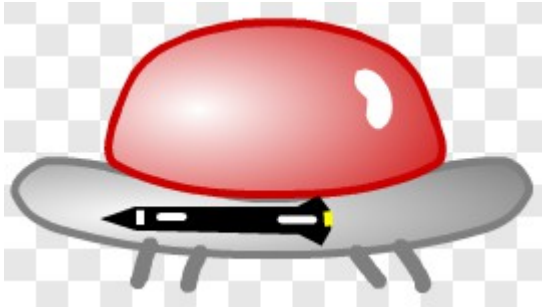


Залейте ракету цветом.



Выглядит угрожающе. Можете ещё немного украсить ракету. У меня получилось вот так.

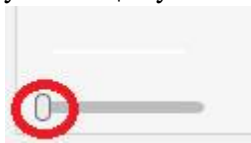
Сыроманское НЛО к боевому вылету готово!



Продолжаем рисовать спрайты. Для создания эффекта космического полёта нам понадобятся три звезды. Как обычно, сначала сделаем одну звезду, запрограммируем её, а потом клонируем.

Выберите «Кисть».

Задайте минимальную толщину линии.

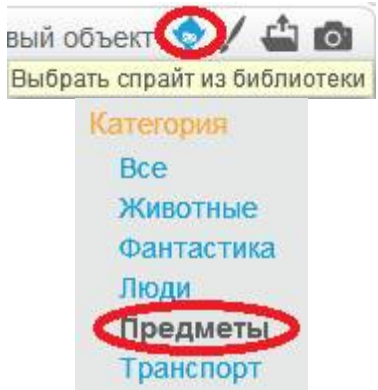


Выберите белый цвет.

Очень сильно увеличьте масштаб.

Поставьте маленькую белую точку точно в центр экрана.

Костюм звёздочки готов. Осталось только добавить буквы для надписи «GAME OVER» (игра окончена). Выберем их из библиотеки спрайтов.




В самом низу можно найти буквы латинского алфавита. Сначала выберите «G».

G-block

Потом выберите «A».

A-block

Потом «M», «E», «O», «V», «E», «R».

M-block

E-block

O-block

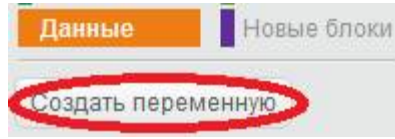
V-block

E-block

R-block

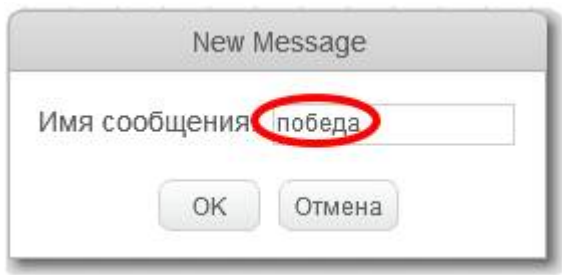
Все спрайты готовы. Начинаем программировать. Выберите сцену.

Сначала создадим глобальную переменную «Очки». Она будет считать подбитых инопланетян.

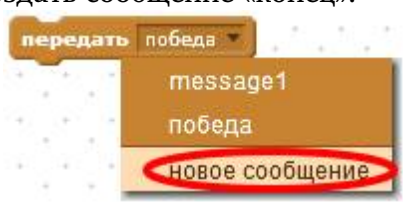



Сделайте программу для сцены. В ней всего два скрипта. Первый скрипт задаёт начальное значение переменной «Очки», и ждёт, пока Кот насшибает 40 НЛО. Когда это произойдёт, он передаст сообщение «победа». Если же Кот не сможет победить захватчиков и погибнет в неравном бою, то будет передано сообщение «конец», при получении которого сцена изменит костюм чёрного космоса на серую заставку. Для того, чтобы создать новое сообщение, кликните на чёрный треугольничек, и выберите из выпадающего списка “новое сообщение».

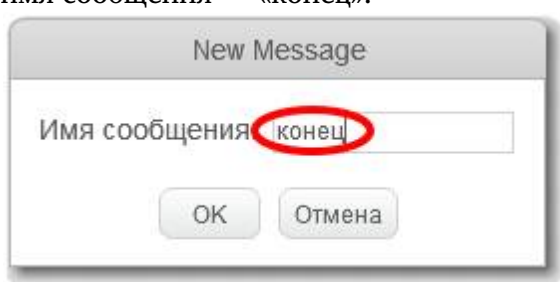
Введите имя нового сообщения — «победа».




Не забудьте создать сообщение «конец».




Введите имя сообщения — «конец».



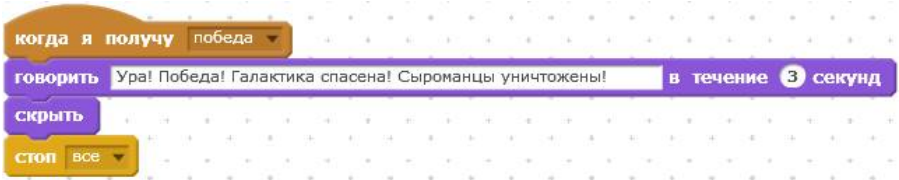

Теперь можно создать два скрипта.

После этого сделайте программу для Кота. В ней будет пять скриптов. Первый скрипт перемещает Кота в левую часть экрана и делает его видимым.

Следующие два скрипта служат для управления движением Кота.

Следующий скрипт отрабатывает сообщение «конец». При получении этого сообщения Кот исчезнет.

Следующий скрипт работает в случае успешного завершения контртеррористической операции. Кот радостно произносит речь, выполнение программы завершается. Галактика может спать спокойно.




После Кота запрограммируем Луч. Его программа имеет два скрипта. Первый скрипт постоянно помещает Луч в точку привязки Кота. Чтобы он вылетал точно из бластера.

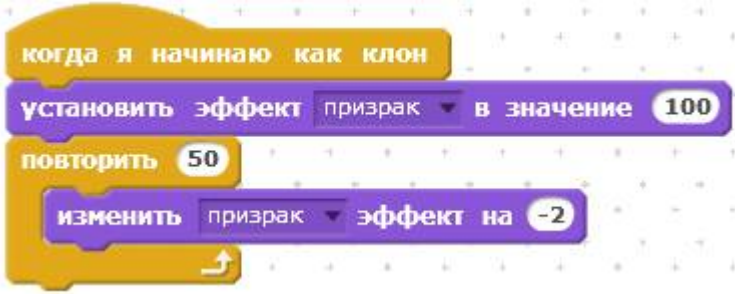
Второй скрипт — скрипт выстрела. При нажатии клавиши Пробел слышен звук, и на одну сотую долю секунды мелькает луч бластера.

Теперь запрограммируем звёздочку. У неё будет пять скриптов. Первый скрипт помещает звёздочку в центр экрана и постоянно создавать её новые клоны.

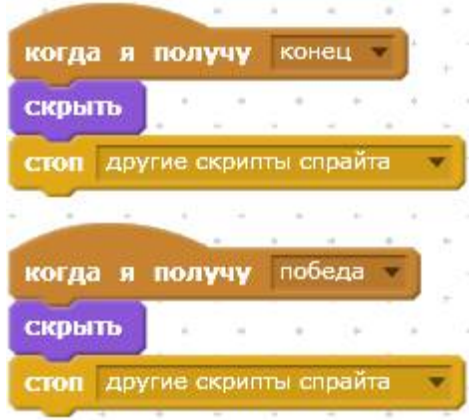
Второй скрипт отправляет всех клонов звёздочки в двухсекундное плавание в неизвестном направлении. Координаты точки назначения X и Y выбираются случайным образом. После того, как клон звёздочки доплывёт до точки назначения, он исчезает.

когда я начинаю как клон
перейти в x: 0 y: 0
плыть 2 секунд в точку x: выдать случайное от -240 до 240 y: выдать случайное от -180 до 180
удалить клон

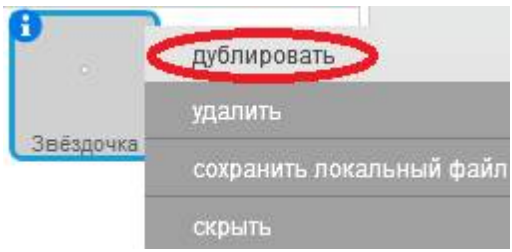
Третий скрипт, при появлении нового клона звёздочки, устанавливает эффект призрак в значение 100, тем самым делая клонов звёздочки невидимыми. Потом значение невидимости плавно изменяется до ноля (если 50 раз отнять 2 от 100, получится 0). При этом будет казаться, что звёзды как будто выплывают из глубин космоса.




Радость победы и горечь поражения бездушные звёзды не испытывают, поэтому при получении подобных сообщений просто спрячутся и остановят работу всех скриптов, перестанут летать.




Давайте протестируем работу программы звёздочки. Нажмите на зелёный флаг. Из цента экрана на Вас полетят звёзды, но, мне кажется, что их маловато. На самом деле, когда Вы летите в космосе, то звёзд гораздо больше. Для того, чтобы добавить в нашу игру побольше звёзд, надо два раза дублировать спрайт звёздочки, звёзд станет в три раза больше. Дублируйте звёздочку два раза.

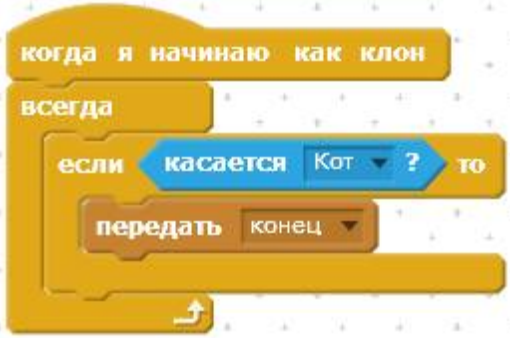



Появились спрайты звёздочка2 и звёздочка3. Скрипты у новых звёздочек такие же, как и у первой. Запустите программу. Звёзд в космосе стало гораздо больше.

Настало время запрограммировать НЛО. У него будет самая сложная программа, в ней будут даже самодельные блоки, позволяющие НЛО двигаться по случайной траектории. Всего у НЛО будет десять скриптов. Первый скрипт.

Второй скрипт.

Третий скрипт.

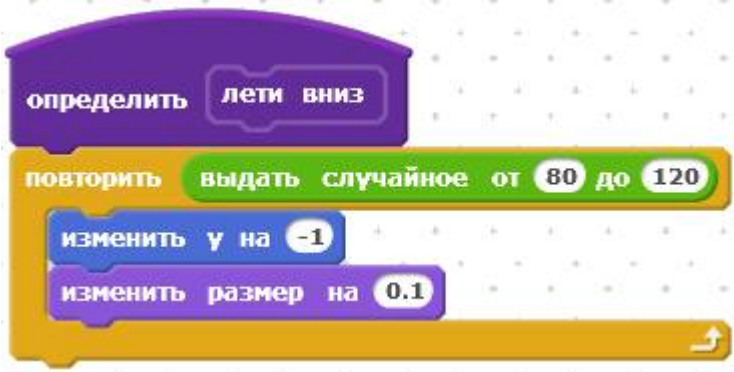



Четвёртый скрипт.

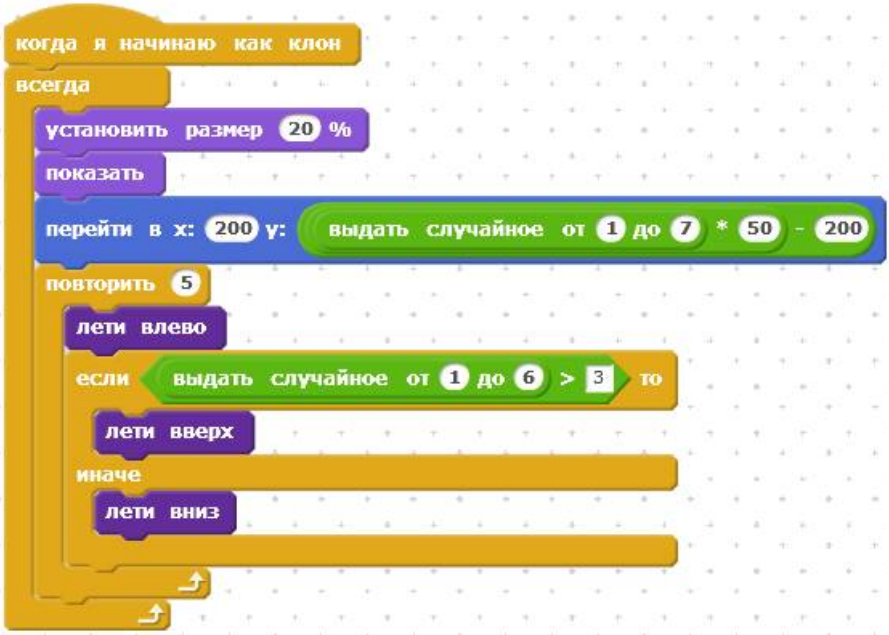
Пятый скрипт.

Шестой скрипт.

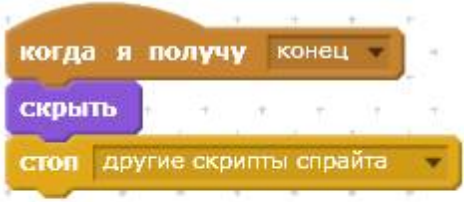
Седьмой скрипт.

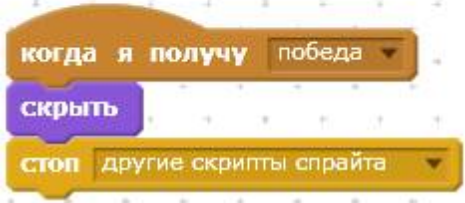



Восьмой скрипт.


когда я начинаю как клон
всегда
установить размер 20 %
показать
перейти в x: 200 y: выдать случайное от 1 до 7 * 50 - 200
повторить 5
лети влево
если выдать случайное от 1 до 6 > 3 то
лети вверх
иначе
лети вниз

Девятый и десятый скрипты.



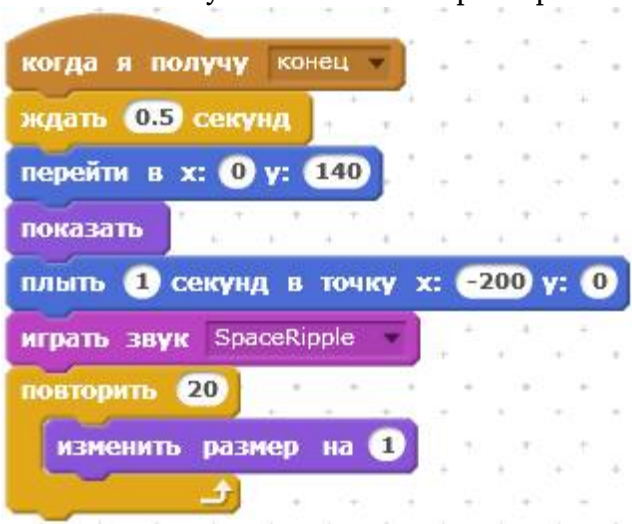




Теперь запрограммируем заставку «GAME OVER», которая появляется в случае проигрыша. У каждой буквы будет по два скрипта. Первый крипт у всех букв одинаковый.

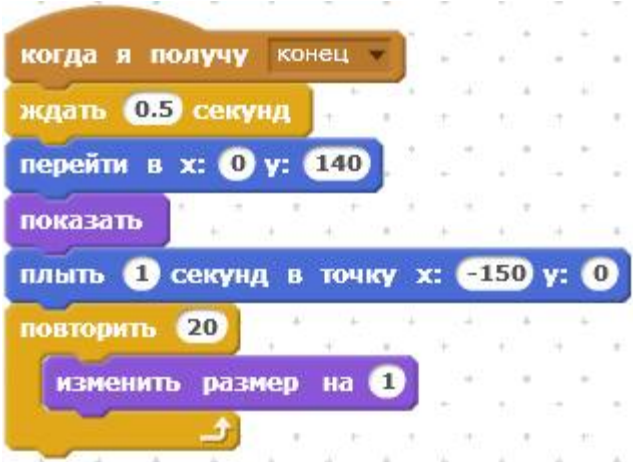
Вторые скрипты у букв почти одинаковые. Отличаются только одним числом — координатой X. Второй скрипт буквы G.

Буква появляется, плывёт в нужную точку в левой части

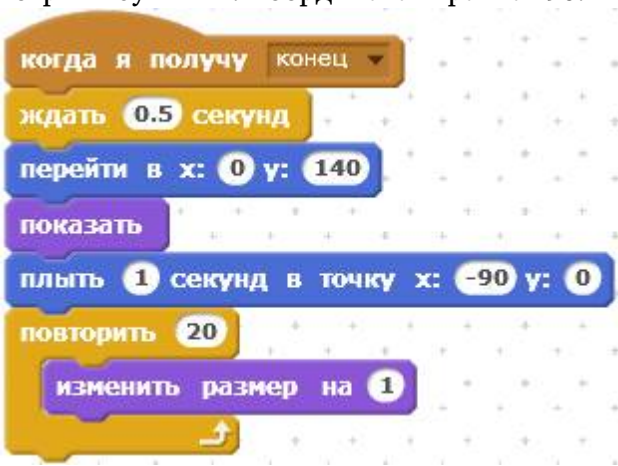
экрана, а потом немного увеличивает свой размер.


когда я получу конец
ждать 0.5 секунд
перейти в x: 0 y: 140
показать
плыть 1 секунд в точку x: -200 y: 0
играть звук SpaceRipple
повторить 20
изменить размер на 1

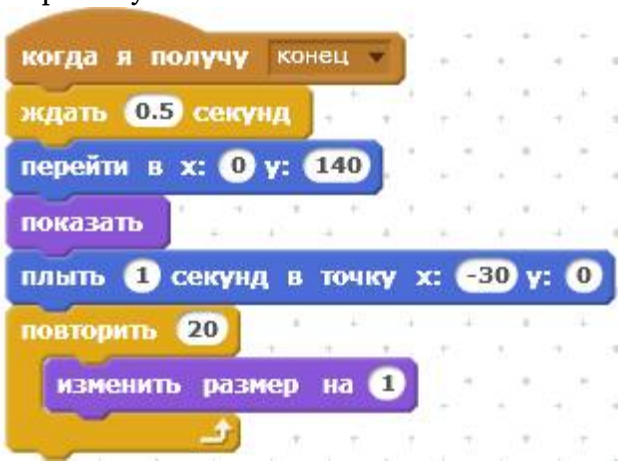
Вот второй скрипт буквы А. Координата X равна -150.


когда я получу конец
ждать 0.5 секунд
перейти в x: 0 y: 140
показать
плыть 1 секунд в точку x: -150 y: 0
повторить 20
изменить размер на 1

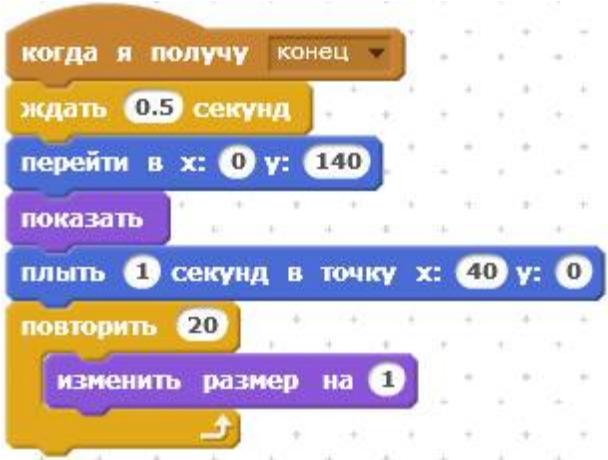
Второй скрипт буквы М. Координата X равна -90.


когда я получу конец
ждать 0.5 секунд
перейти в x: 0 y: 140
показать
плыть 1 секунд в точку x: -90 y: 0
повторить 20
изменить размер на 1

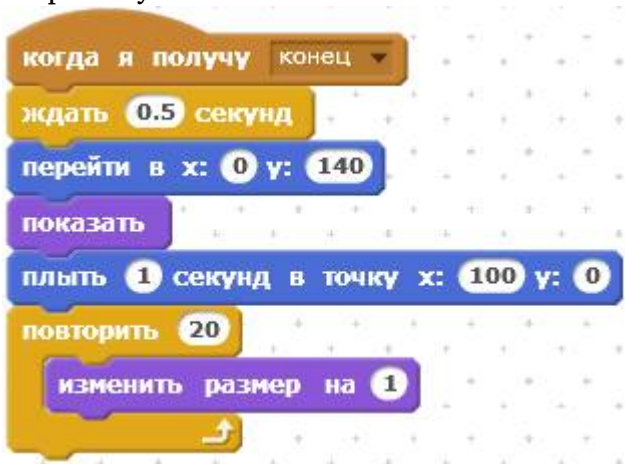
Второй скрипт буквы Е.


когда я получу конец
ждать 0.5 секунд
перейти в x: 0 y: 140
показать
плыть 1 секунд в точку x: -30 y: 0
повторить 20
изменить размер на 1

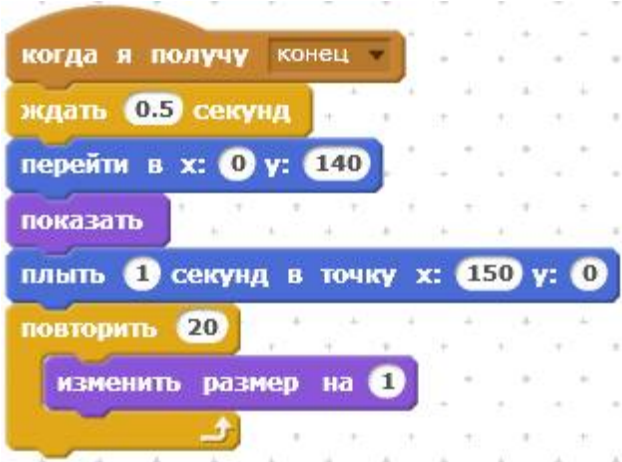
Второй скрипт буквы О.


когда я получу конец
ждать 0.5 секунд
перейти в x: 0 y: 140
показать
плыть 1 секунд в точку x: 40 y: 0
повторить 20
изменить размер на 1

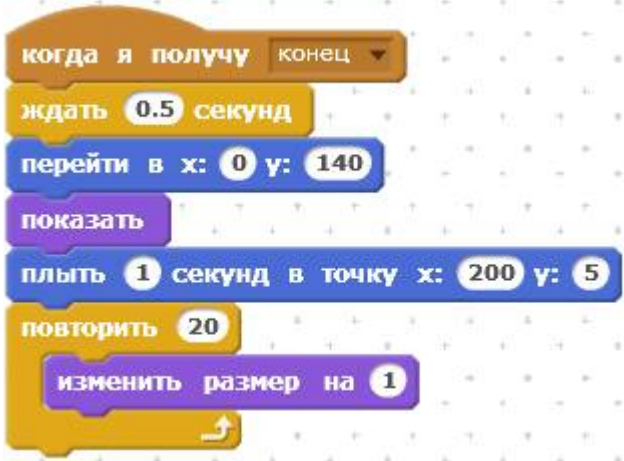
Второй скрипт буквы V.


когда я получу конец
ждать 0.5 секунд
перейти в x: 0 y: 140
показать
плыть 1 секунд в точку x: 100 y: 0
повторить 20
изменить размер на 1

Второй скрипт второй буквы Е.


когда я получу конец
ждать 0.5 секунд
перейти в x: 0 y: 140
показать
плыть 1 секунд в точку x: 150 y: 0
повторить 20
изменить размер на 1

Второй скрипт второй буквы Е.


когда я получу конец
ждать 0.5 секунд
перейти в x: 0 y: 140
показать
плыть 1 секунд в точку x: 200 y: 5
повторить 20
изменить размер на 1

Игра готова. Запустите её и не дайте врагам захватить галактику!

Вопросы. 1. Кто первым из жителей нашей планеты отправился на разведку в космос?

2. Как ускорить полёт НЛО?

3. Слышны ли в космосе взрывы вражеских ракет?

Задания к игре «Космическая битва».

1. Сделайте заставку в начале игры. На заставке должно быть написано Ваше имя и название игры.

2. Сделайте так, чтобы после 10 выстрелов требовалось 2 секунды на перезарядку оружия.

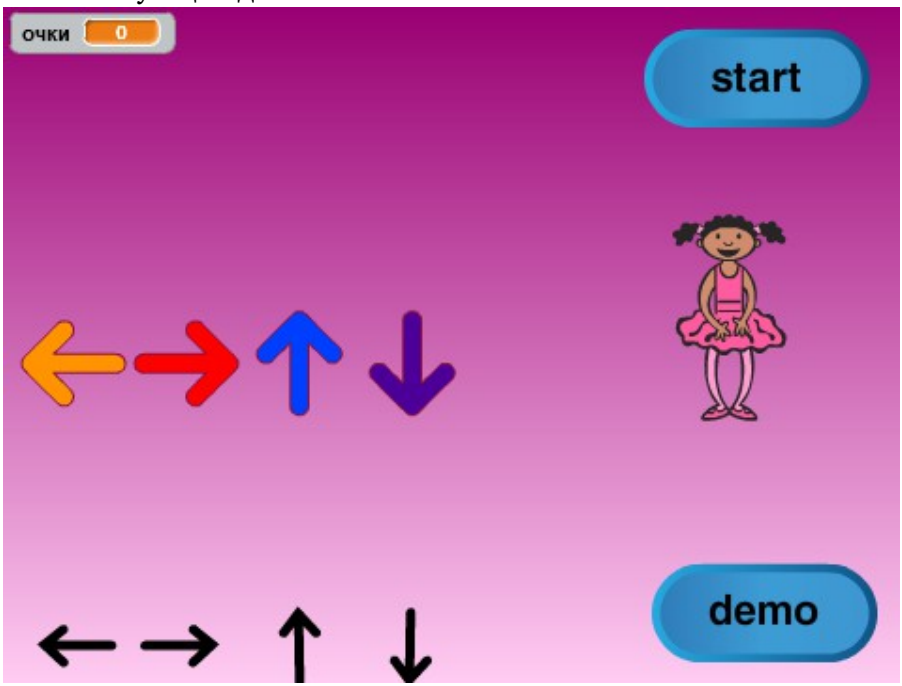
3. Для того, чтобы игра не заканчивалась сразу, сделайте Коту 5 жизней.

# ГЛАВА 6. ТАНЦЕВАЛЬНЫЙ КОВРИК.

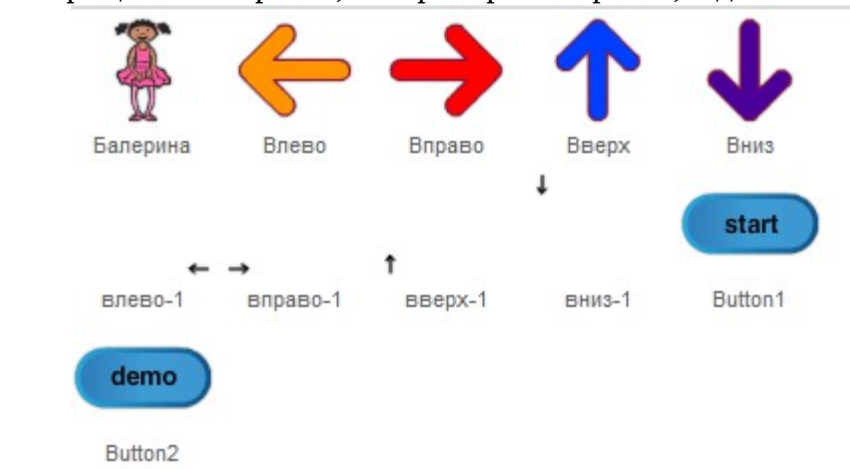
Жила была девочка, которая очень любила танцевать. На день рождения ей подарили танцевальный коврик. Девочка была так рада, что танцевала весь день напролёт. Она нажимала все стрелки в правильной последовательности, и набрала больше

всех очков! Сколько очков сможете набрать Вы? Главное — не нажимать кнопку demo!

В этой игре чёрные стрелочки по одной плывут вверх и соприкасаются с цветными. Вам предстоит нажимать на клавиатуре на такие стрелки, которые в этот момент соприкасаются и подсвечиваются на экране. За каждое правильное нажатие начисляется по одному очку. При нажатии на клавиатуре стрелочек, девочка-балерина будет совершать соответствующие движения.

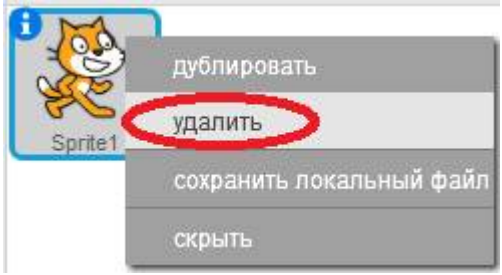



В этой игре будет всего одиннадцать спрайтов. Балерина, четыре цветных стрелки, четыре чёрных стрелки, и две кнопки.

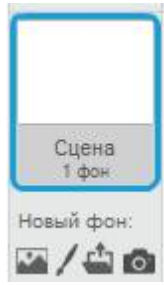



Создание игры начинаем с рисования сцены и спрайтов.

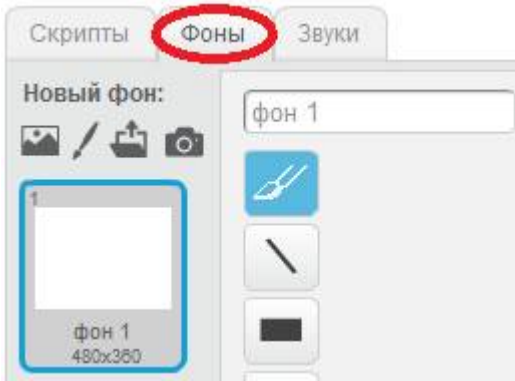
Первым делом удалите «Sprite1»



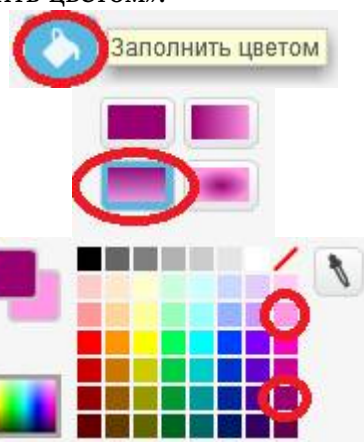

Выберите сцену.




Перейдите на закладку Фоны.

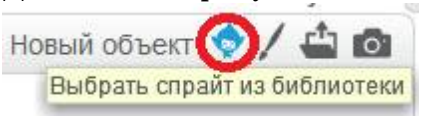



Залейте фон градиентной (двухцветной) заливкой. Выберите инструмент «Заполнить цветом».

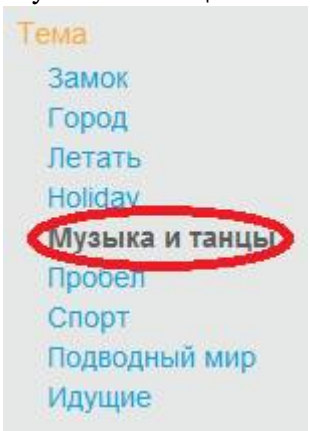


Получится вот такой красивый фон.

Сцена готова. Добавьте Балерину из библиотеки спрайтов.



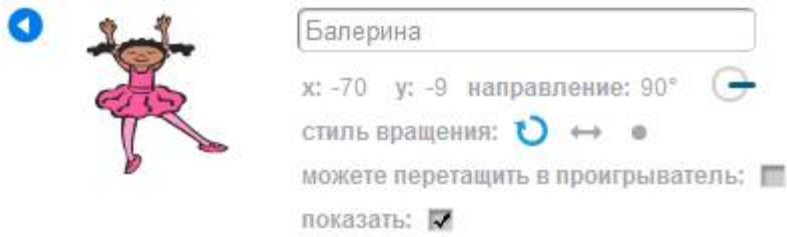

Выберите раздел «Музыка и танцы».



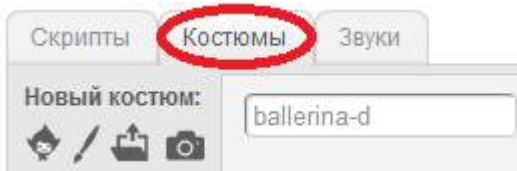

Выберите нашу красавицу и нажмите «Ок».

Переименуйте её в Балерину. Разверните окно свойств спрайта.

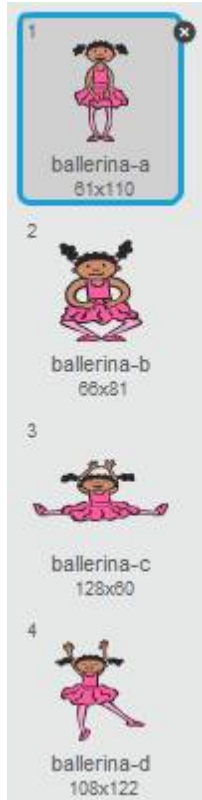
Введите новое имя.




Перейдите на вкладку «Костюмы».

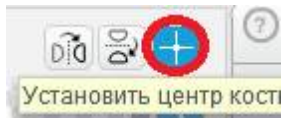



У Балерины уже есть четыре костюма! Как раз, по одному костюму для каждой стрелки.

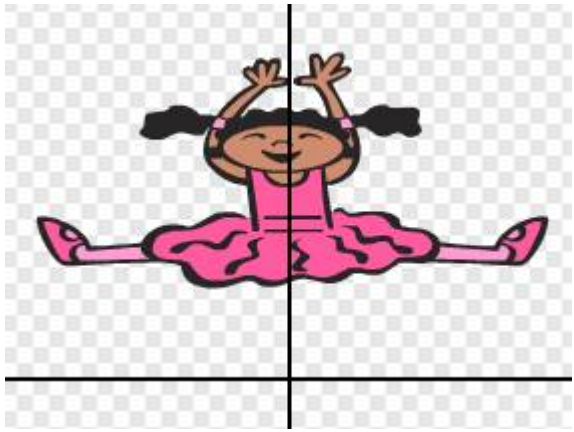



Третий костюм, в котором Балерина сидит на шпагате, мы переделаем на прыжок, который будет происходить при нажатии на клавиатуре стрелки вверх.

Выберите третий костюм и нажмите кнопку «Установить центр костюма».

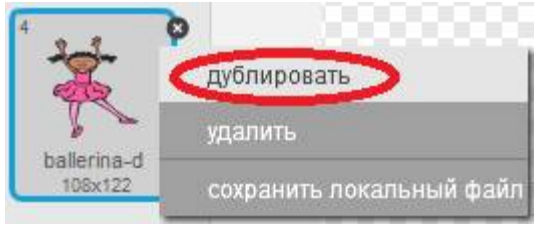



Установите центр костюма ниже Балерины. Теперь она будет делать шпагат в прыжке!

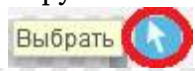


Балерине не хватает ещё одного костюма — костюма, где она поднимает правую ногу. Этот костюм она будет надевать, когда нажата клавиша «стрелка влево» на клавиатуре.

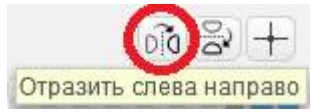
Дублируйте костюм «ballerina-d».




Выделите Балерину инструментом выделения.

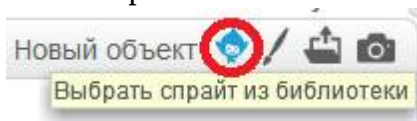


А теперь, нажмите вверху кнопку «Отразить слева направо».

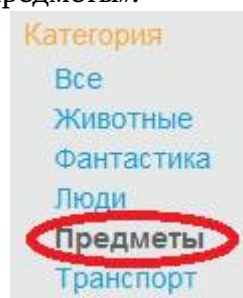



Балерина зеркально отразится.

Балерина зеркально отразится.

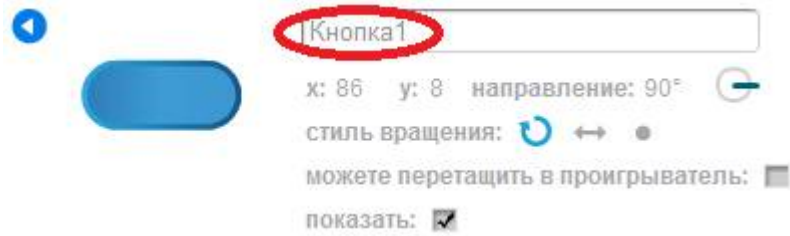



Выберите раздел «Предметы».



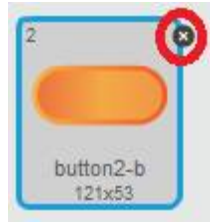

Выберите кнопку «Button2» и нажмите «Ок».

Переименуйте кнопку.




Перейдите на вкладку «Костюмы».

Удалите второй костюм кнопки, он нам не понадобится.

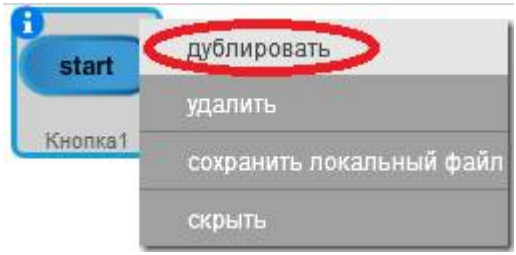



Теперь нам нужно написать на кнопке слово «start». Эта кнопка будет запускать программу. Выберите инструмент «Текст».

Кликните в кнопку и напишите «start».

Если у Вас, как и у меня, получится криво, возьмите инструмент «Выбрать», и переместите надпись точно в центр.

Кнопка готова. Дублируйте её.

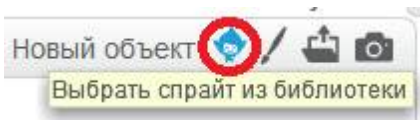



У появившейся кнопки сразу будет правильное имя «Кнопка2».

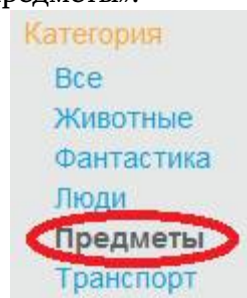
Отредактируйте костюм Кнопки2. Выберите инструмент «Текст».

Кликните в надпись, и измените её.

Кнопки готовы. Теперь добавим первую стрелку. После того, как запрограммируем её поведение, будем три раза дублировать её, чтобы сразу получились стрелки со скриптами. Останется только немного изменить скрипты. Откройте библиотеку спрайтов.



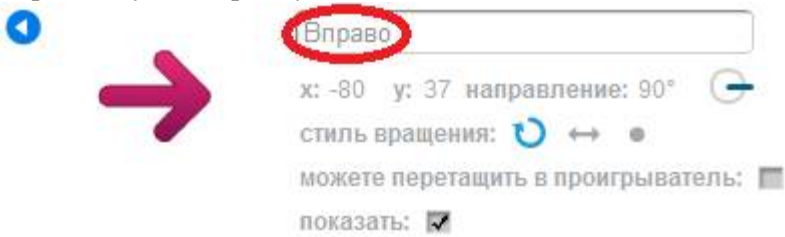

Выберите раздел «Предметы».




Выберите стрелку «Arrow1» и нажмите «Ок».

Arrow1

Переименуйте стрелку.

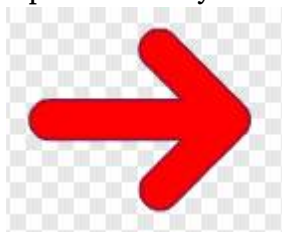



Перейдите на вкладку «Костюмы».

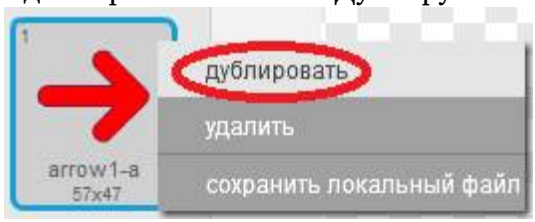
Удалите все костюмы, кроме первого, они нам не понадобятся.

Стрелку надо окрасить красным цветом. Выберите инструмент «Окрасить форму».

Вот такая красная стрелочка получится.

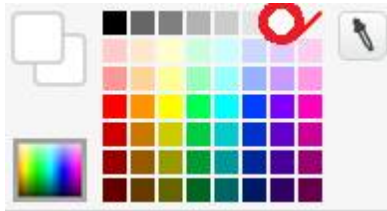


При касании красной стрелки чёрной стрелочкой, она должна подсветиться. Для этого необходимо сделать второй костюм, и у него залить ободок стрелочки белым. Дублируйте первый костюм.




Для облегчения заливки ободка увеличьте масштаб.

Выберите беленький цвет.



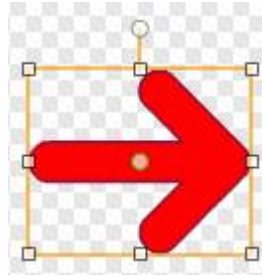
Аккуратно коснитесь инструментом окраски границы стрелочки. Она станет белой, и почти невидимой.

У нас получилось два костюма стрелочки Вправо.

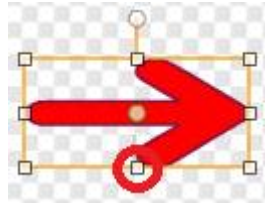
Теперь надо сделать чёрную стрелочку. Давайте сделаем её из стрелки Вправо. Дублируйте стрелочку Вправо. У неё сразу появилось неплохое имя - Вправо2, оставим его. Зайдите на вкладку Костюмы стрелочки Вправо2, и удалите второй костюм.

Оставшуюся стрелочку выделите с помощью инструмента «Выбрать».

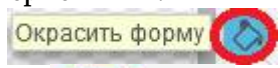
Стрелочку будет обведена рамкой.

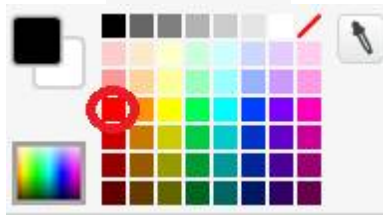


Тащите нижний центральный квадратик выделения немного вверх. Стрелочка станет тоньше.



Теперь залейте её чёрненьким.

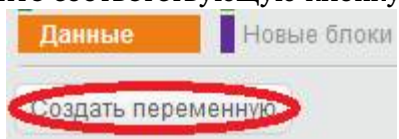




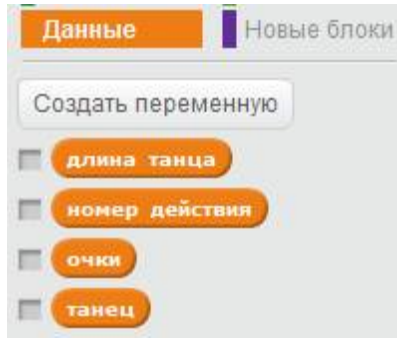
Вот такая стрелочка получится.

Сейчас, когда все костюмы нарисованы, начинаем создавать программу.

Сначала сделаем программу сцене. Начнём с создания переменных, нажмите соответствующую кнопку.

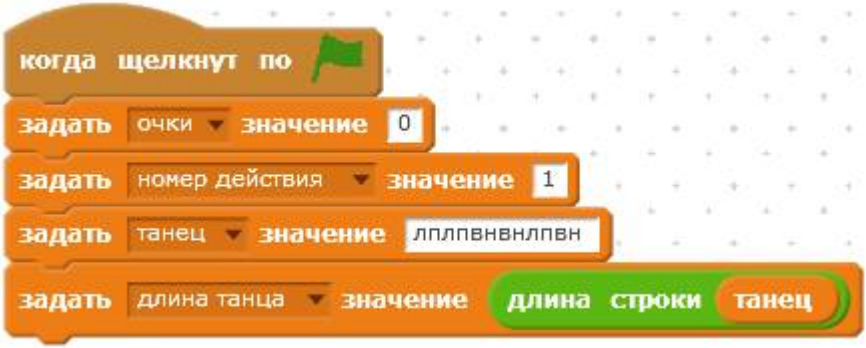



Создайте четыре глобальные переменные: «танец», «очки», «длина танца», и «номер действия».

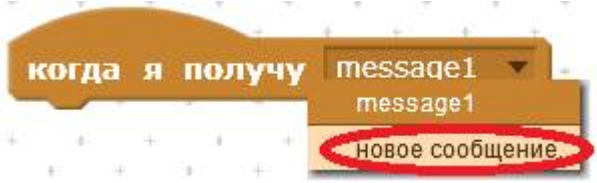


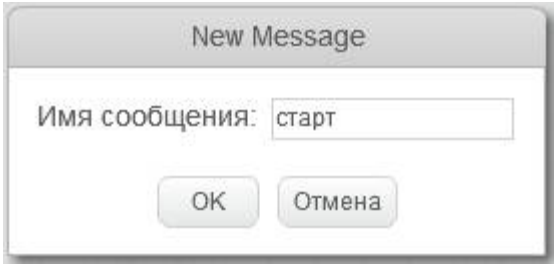

Переменная очки будет подсчитывать количество правильных нажатий стрелок на клавиатуре. В переменной танец закодирована очерёдность движения маленьких стрелок, то есть, какая стрелка следует за какой, это как бы очередь. Стрелки закодированы буквами. Л — двигается стрелка Влево. П — двигается стрелка Вправо. В — стрелка Вверх. Н — стрелка Вниз. Для того, чтобы определить сколько всего надо совершить действий, служит переменная «длина танца», а для определения номера действия служит переменная «номер действия».

Всего у сцены будут два скрипта. Сделайте первый скрипт, он устанавливает начальные значения переменных.


когда щелкнут по
задать очки значение 0
задать номер действия значение 1
задать танец значение лплпвнвнлпвн
задать длина танца значение длина строки танец

Второй скрипт сцены каждую четверть секунды передаёт разные сообщения в соответствии с последовательностью, закодированной в переменной «танец». Сначала создайте новое сообщение «старт».





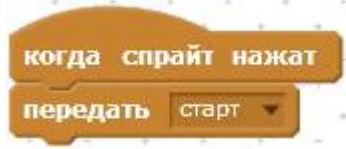

Сделайте скрипт. В процессе создания Вам будет нужно создать ещё четыре сообщения.

когда я получу старт
повторить длина танца
если буква номер действия в танец = л то
передать влево
ждать 0.25 секунд
если буква номер действия в танец = п то
передать вправо
ждать 0.25 секунд
если буква номер действия в танец = в то
передать вверх
ждать 0.25 секунд
если буква номер действия в танец = н то
передать вниз
ждать 0.25 секунд
изменить номер действия на 1
передать конец

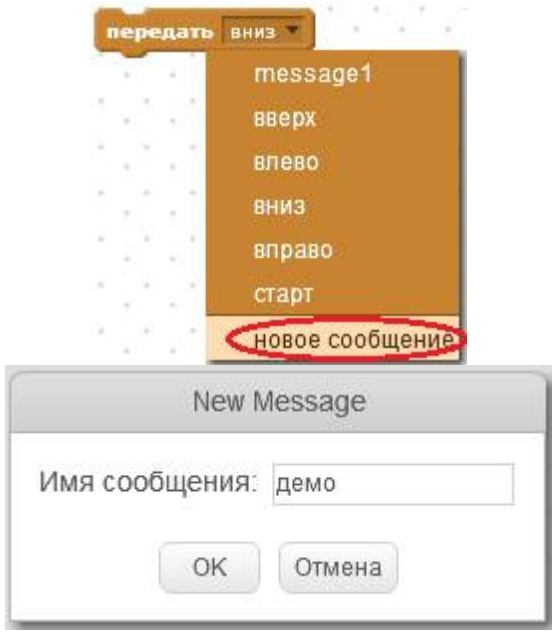
Рассмотрим, как работает этот скрипт. В самом начале работы программы значение переменной «номер действия» равно единице, значит, первое условие будет выглядеть так.

Если первая буква в танце равна «л», а она равна, то передаём сообщение «влево». Получив это сообщение, маленькая стрелочка Влево2 поплывет вверх. Скрипт работает циклически, он повторяется ровно столько раз, сколько букв в переменной «танец», и в каждом цикле изменяет значение переменной «номер действия» на единицу. Это значит, что после сообщения «влево», соответствующего первой букве, будет передано сообщение, соответствующее второй букве в переменной «танец», а это «вправо».

Теперь запрограммируйте кнопки. У Кнопки1 только один скрипт, он просто передаёт сообщение «старт».

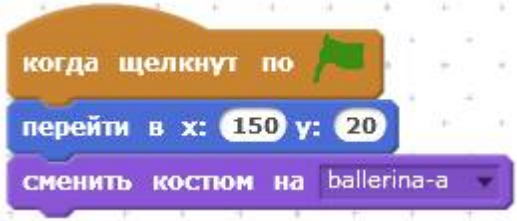



У Кнопки2 тоже всего один скрипт, он передаёт сообщение «демо». Создайте это сообщение.

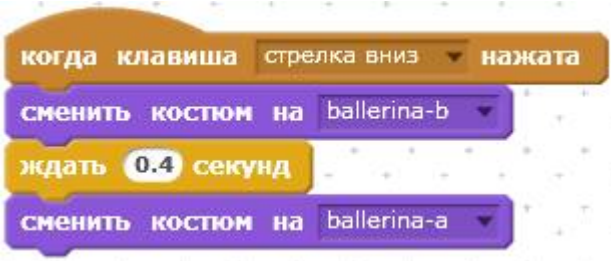



Сделайте скрипт Кнопки2.

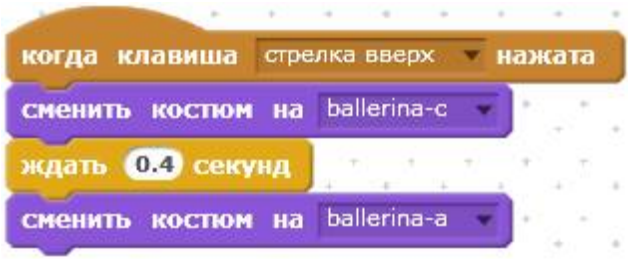
Приступаем к созданию программы для Балерины. Её программа состоит из восьми скриптов. Первый скрипт помещает Балерину в правую часть экрана, и надевает ей первый костюм.



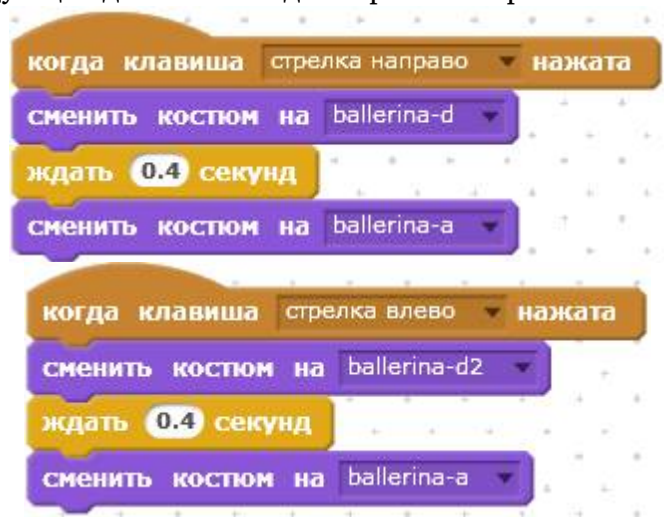

Следующие четыре скрипта определяют, как Балерина реагирует на нажатие стрелок на клавиатуре. Если на клавиатуре нажата клавиша «стрелка вниз», по Балерина сменит костюм на приседающий, и вернётся в исходное положение.



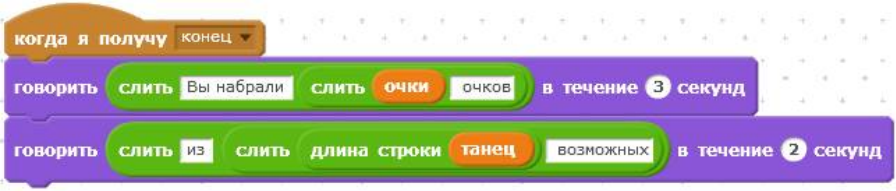

Если на клавиатуре нажата клавиша «стрелка вверх», по Балерина сменит костюм на подпрыгивающий, и вернётся в исходное положение.




Следующие два костюма для стрелок направо и влево.

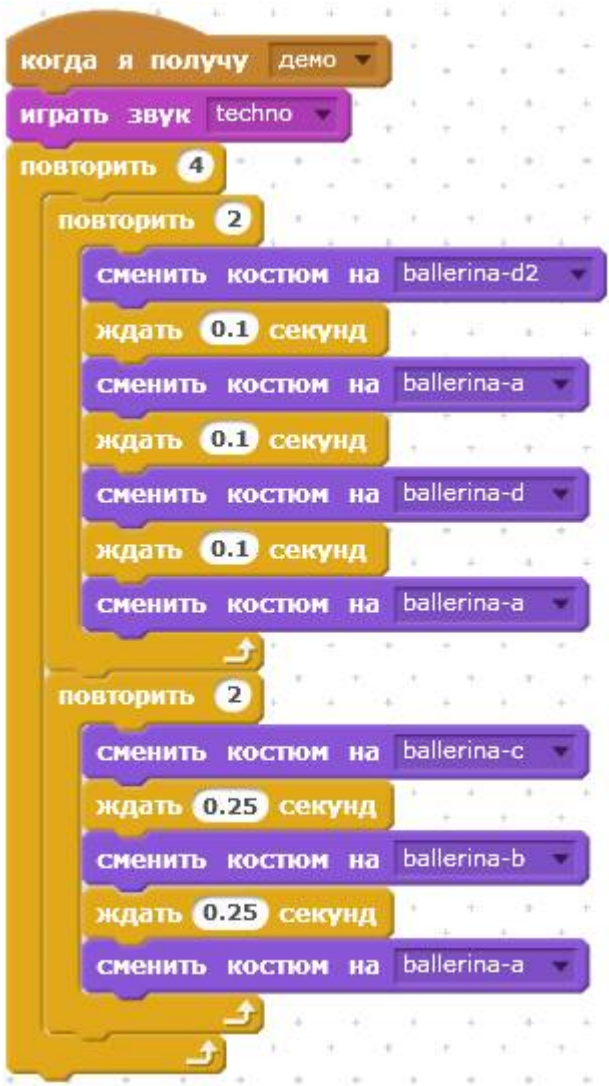

когда клавиша стрелка направо нажата
сменить костюм на ballerina-d
ждать 0.4 секунд
сменить костюм на ballerina-a
когда клавиша стрелка влево нажата
сменить костюм на ballerina-d2
ждать 0.4 секунд
сменить костюм на ballerina-a

С помощью следующего скрипта девочка в конце игры скажет сколько очков Вы набрали.

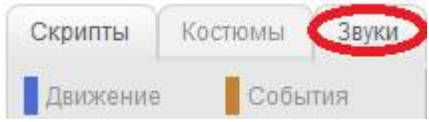

когда я получу конец
говорить слить Вы набрали слить очки очков в течение 3 секунд
говорить слить из слить длина строки танец возможных в течение 2 секунд

Вот небольшая подсказка, как собрать такой сложный скрипт.

Следующий скрипт будет заставлять девочку танцевать демонстрационный танец. Сделайте девочке такой скрипт.

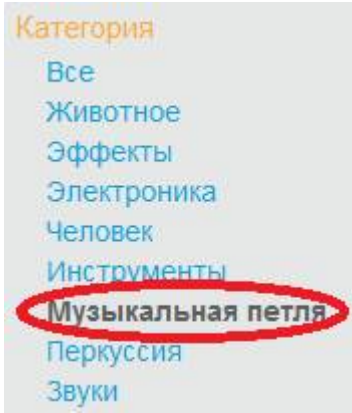



Звук «techno» сможете найти в библиотеке звуков. Перейдите на закладку «Звуки».




Нажмите на кнопку библиотеки звуков.

Перейдите в раздел «Музыкальная петля».

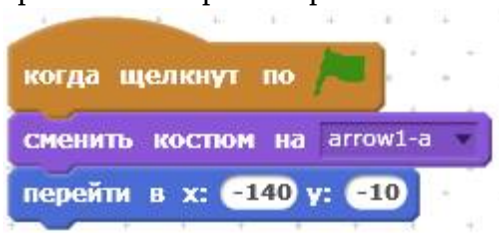



Выберите требуемый звук, и нажмите «Ок».

Теперь, при получении от Кнопки2 сообщения «демо», Балерина пустится в пляс, посмотрите, как её колбасит!

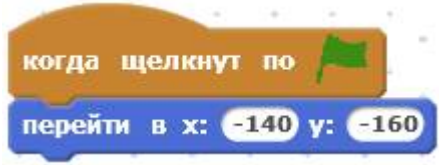
Можно начинать программировать стрелки. Последовательность действий (алгоритм) будет такой. Сначала делаем программы для стрелки Вправо и для маленькой стрелки Вправо2. Потом с помощью дублирования красной стрелки получим стрелки Влево, Вверх и Вниз, а с помощью дублирования маленькой стрелки Вправо2, получим маленькие стрелки Влево2, Вверх2 и Вниз2.

В первую очередь сделаем программу для красной стрелки. В ней всего два скрипта. Вот первый скрипт.



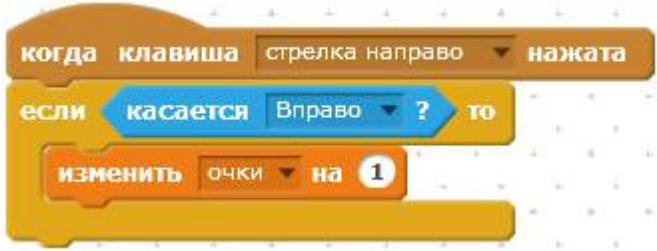

Вот второй скрипт.

Теперь сделаем программу маленькой стрелке Вправо2. У неё всего три скрипта. Первый скрипт устанавливает стрелку в начальное положение в центре экрана.

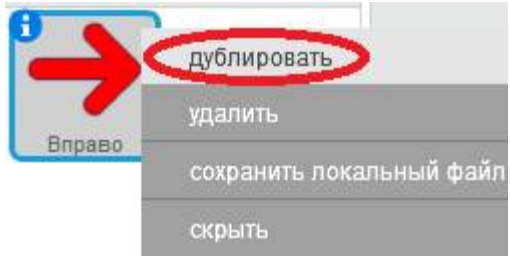



Второй скрипт перемещает стрелочку Вправо2 50 раз на 3 точки вверх — всего на 150 точек вверх, и снова возвращает в исходную точку.

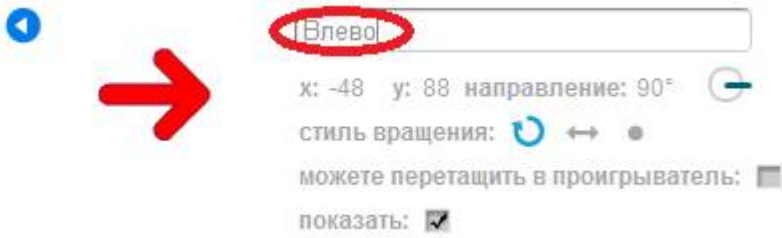
Третий скрипт проверяет, успели ли Вы нажать на стрелку Вправо в момент касания стрелок. Если у Вас хорошая реакция, и Вы успели, то переменная «очки» изменится на единицу.



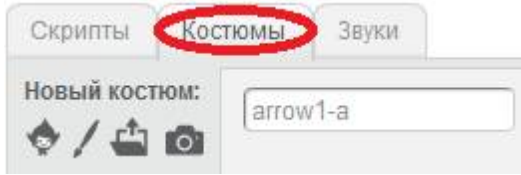

Обе стрелки вправо готовы. Теперь сделаем остальные три больших стрелки путём дублирования красной стрелки. После того, как все стрелки будут готовы, изменим их программы. Сначала сделаем стрелку Влево. Дублируйте красную стрелку.



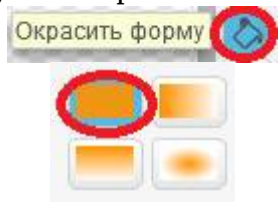

Переименуйте полученный спрайт.

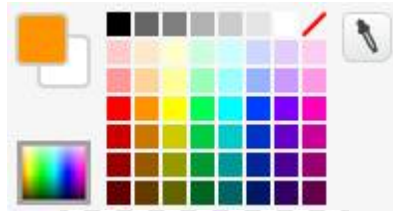



Перейдите на вкладку «Костюмы».

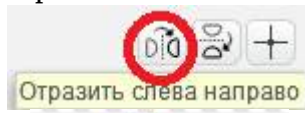



Надо перекрасить стрелку в оранжевый цвет и развернуть влево. Выберите инструмент окраски.



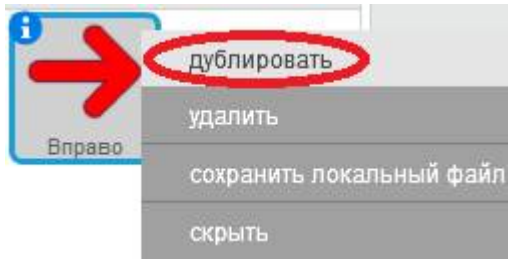


Отразите стрелку зеркально.

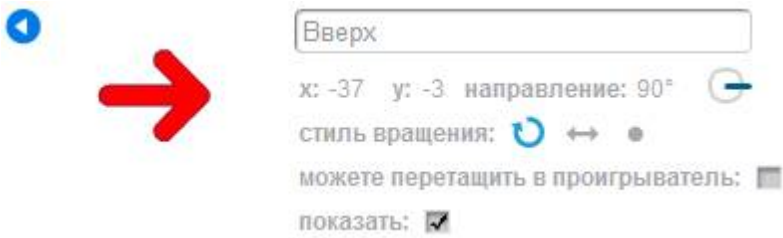


Выберите второй костюм стрелки. Аналогичным образом покрасьте её и отразите.

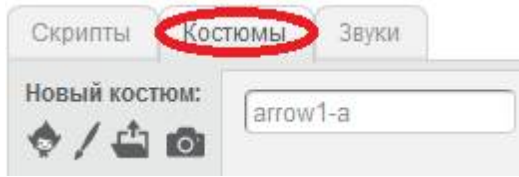
Теперь сделаем стрелку Вверх. Снова дублируйте красную стрелку.




Переименуйте полученный спрайт.



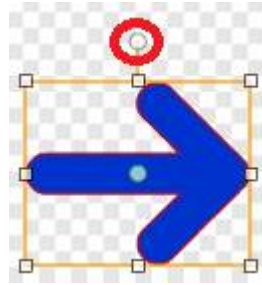

Перейдите на вкладку «Костюмы».



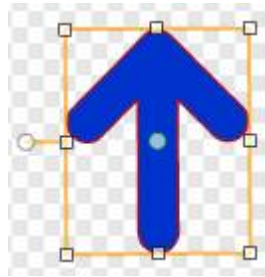

Надо перекрасить стрелку в синий цвет и развернуть вверх. Выберите инструмент окраски.

Окрасьте стрелку.

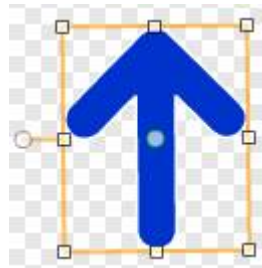
Для того, чтобы повернуть стрелку вертикально, выберите её инструментом выделения.



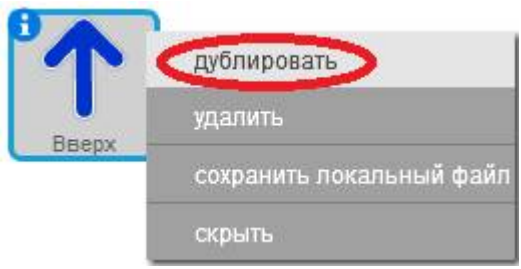
Кликните на верхний кружочек, и поверните стрелку по направлению вверх.



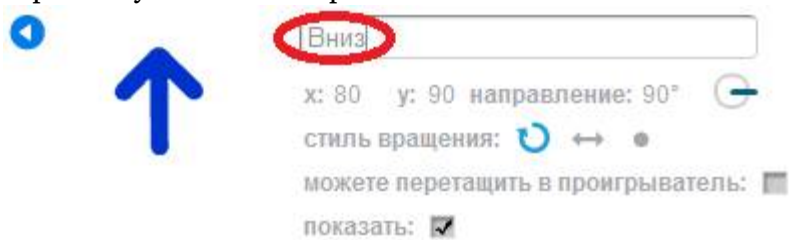
Второй костюм стрелки покрасьте и поверните таким же образом.



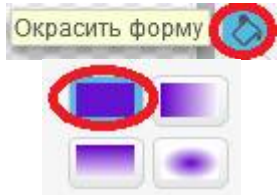
Стрелку вниз создадим дублируя стрелку Вверх.




Переименуйте новый спрайт.



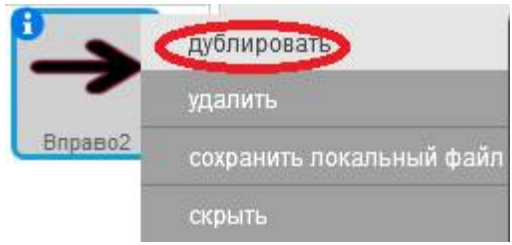

Перекрасьте стрелку в фиолетовый цвет.



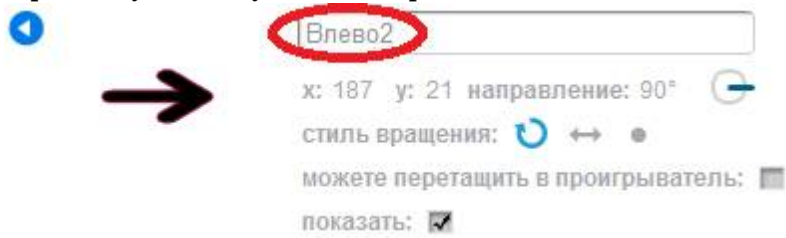
Отразите её зеркально.

Аналогичным образом перекрасьте второй костюм стрелки Вниз, и отразите его зеркально.

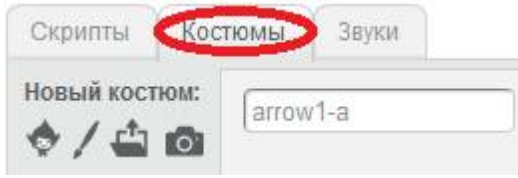
Все большие стрелки нарисованы. Их программы изменим после того, как сделаем все маленькие стрелки. Дублируйте маленькую стрелку Вправо2.



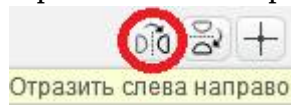

Переименуйте полученный спрайт.




Перейдите на вкладку «Костюмы».

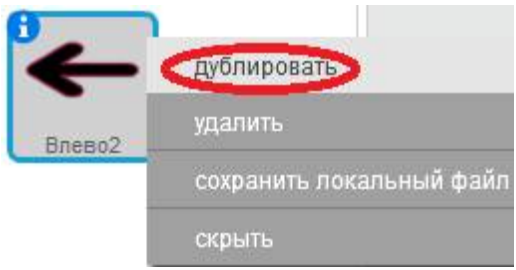



Нажмите кнопку «Отразить слева направо».

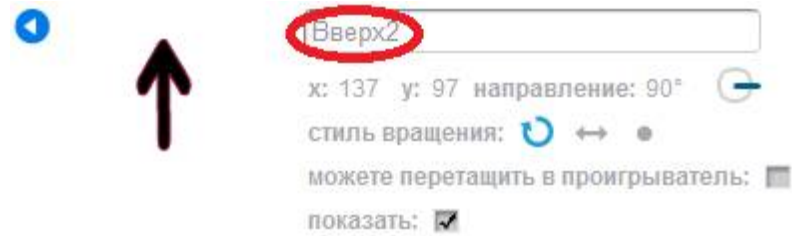



Стрелочка Влево2 готова.

Теперь из неё сделаем стрелочку Вверх2. Дублируйте стрелку Влево2.

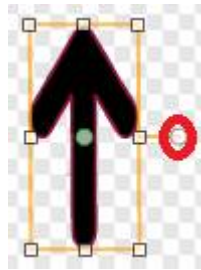



Переименуйте её.

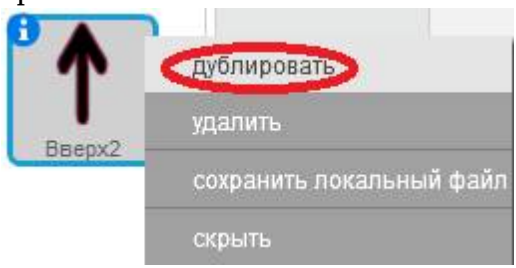



Для того, чтобы повернуть стрелку вертикально, выберите её инструментом выделения.

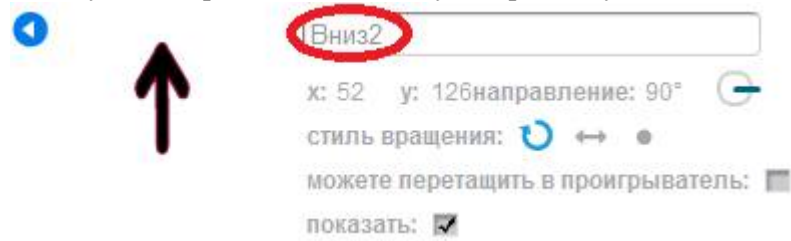
Теперь аккуратно кликните кружочек и поверните стрелку кончиком вверх.



Стрелочка Вверх2 готова. Сделаем из неё стрелочку Вниз2. Дублируйте Вверх2.

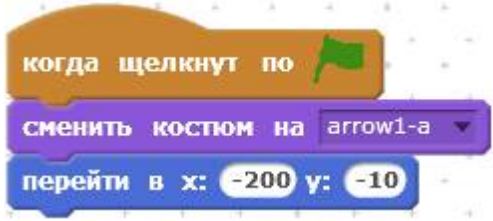



Не забудьте переименовать новую стрелочку.



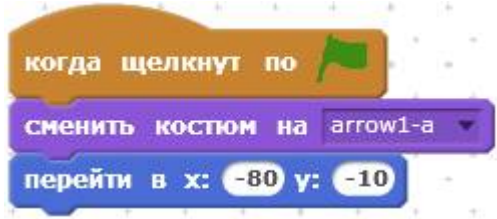

Отразите её зеркально.

Вот теперь, наконец-то, готовы все стрелки. Осталось немного изменить их скрипты, и игра готова. Начните со стрелки Влево. В первом скрипте исправьте координаты X и Y. Первый скрипт должен быть таким.



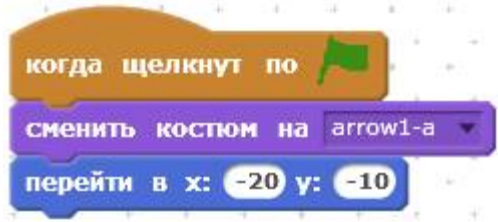

Во втором скрипте исправьте только номер ноты и кого касается. Второй скрипт должен быть таким.

Аналогично исправьте скрипты стрелки Вверх. Вот её первый скрипт.



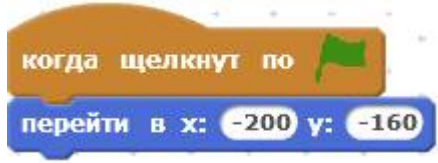

Вот второй скрипт.

Скрипты стрелки Вниз. Первый скрипт.



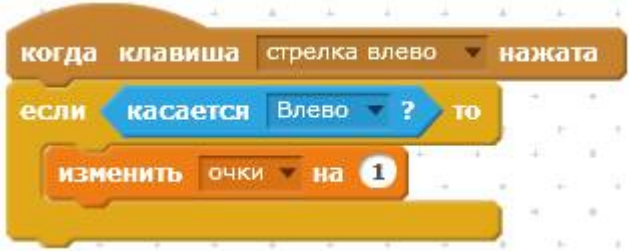

Второй скрипт.

Переходим к маленьким стрелочкам. Вот первый скрипт стрелочки Влево2.




Второй скрипт стрелочки Влево2.

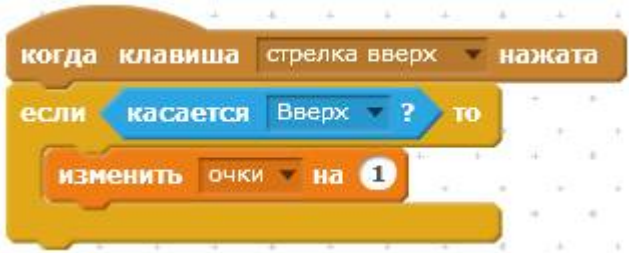
Третий скрипт стрелочки Влево2.




Теперь исправим скрипты стрелочки Вверх2. Первый скрипт.

Второй скрипт.

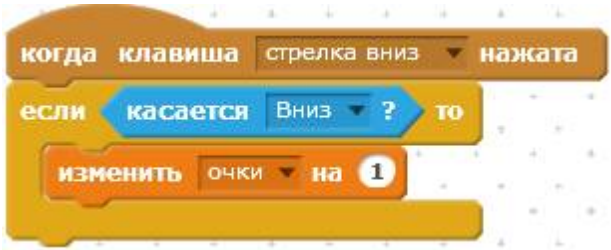
Третий скрипт.




Теперь принимаемся за стрелочку Вниз2. Первый скрипт.

Второй скрипт.

Третий скрипт.




Игра готова.

Вопросы. Вы умеете танцевать?

Задание

1. Сделайте так, чтобы, когда маленькие стрелочки стоят внизу и не двигаются, они были невидимыми. Появляться они должны только тогда, когда двигаются вверх.

2. Удлините танцевальное задание в переменной «танец».

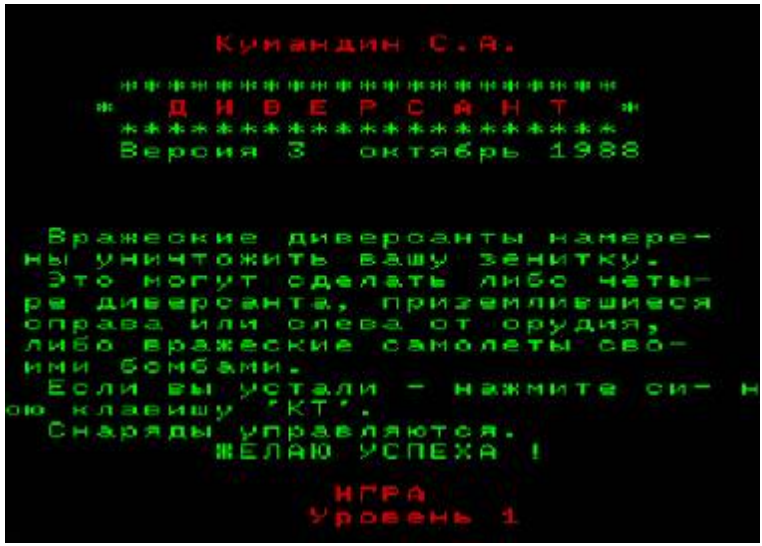
3. загрузите свою mp3 мелодию в скрипт demo и сделайте прикольный танец для девочки.

4. Нарисуйте вместо девочки своего персонажа, который будет танцевать.

5. Запишитесь на танцевальный кружок и научитесь танцевать.

# ГЛАВА 7. ДИВЕРСАНТ.

Эта игра появилась очень давно. В 1988г. её сделал С.А. Кумандин. Правила игры коротко изложены на рисунке. Это фотография экрана древнего компьютера БК-0010.




Здесь написано следующее. Диверсант, Версия 3. Октябрь 1988 г. Вражеские диверсанты намерены уничтожить Вашу зенитку. Это могут сделать либо четыре диверсанта, приземлившиеся справа или слева от орудия, либо вражеские самолёты своими бомбами. Если Вы устали, нажмите синюю клавишу «КТ». Снаряды управляются. Желаю успеха!

Игра работала только на древнем компьютере БК-0010.

Он стоил столько же сколько и автомобиль! Обратите внимание на клавиатуру!

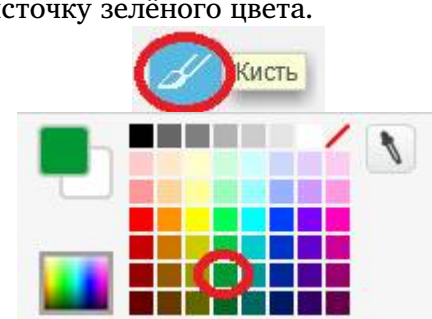
Сколько на ней клавиш? А сколько клавиш на современной клавиатуре? А для чего нужна клавиша Scroll Lock на современной клавиатуре?

Точно такую же игру мы делать не будем, а сделаем упрощенный вариант, в котором Диверсанты будут прыгать по одному. Внизу будет расположена база с Зениткой, за которую играете Вы. Зенитка поворачивается стрелками «вправо» и «влево». Сверху будет летать Самолет, из которого выпрыгивают Диверсанты, бросать бомбы Самолёт не будет. Диверсанты приземляются и двигаются по направлению к Вашей базе. Если Диверсант дойдёт до Базы, то она взлетит на воздух - конец игры. Если Вы попадёте в Диверсанта, то он исчезнет. Если Ваш снаряд попадёт в Самолет, то на нем появится пробоина. Всего у Самолета 5 жизней. Если Вы 5 раз попадёте в Самолет, то он исчезнет - Вы победили.

Запустите Scratch.

Первым делом нарисуем фон. Выберите сцену. Выберите закладку Фоны.

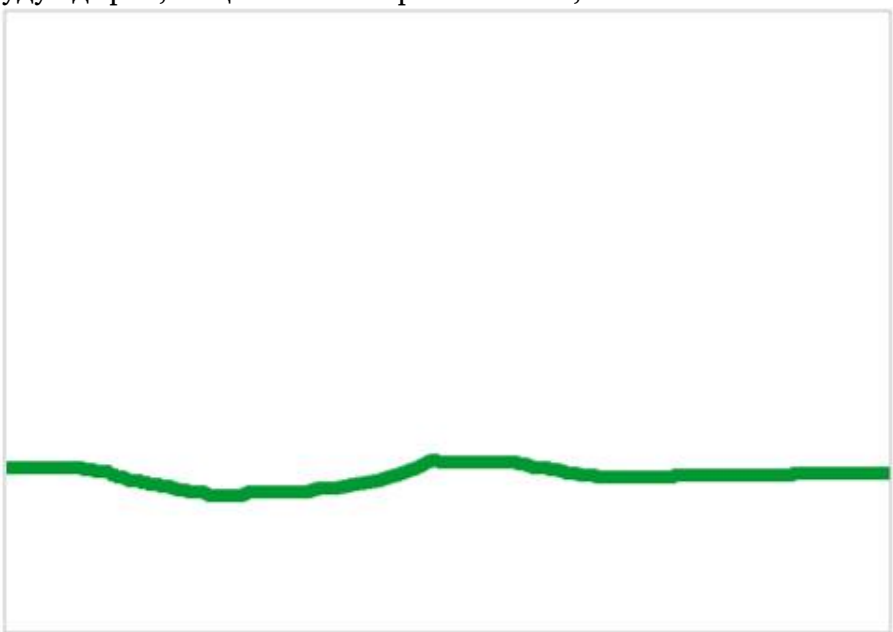
Выберите Кисточку зелёного цвета.




Выберите среднюю толщину линии.

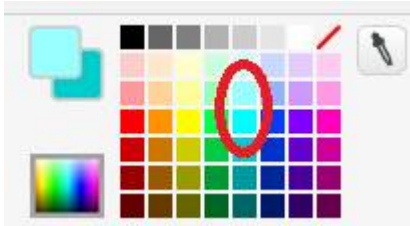
Нарисуйте зеленый горизонт от центра экрана к его краю.

Нарисуйте вторую половину горизонта. Очень важно, чтобы получилась сплошная линия от края до края фона. Потом мы будем использовать инструмент Заливка, и, если в горизонте будут дырки, то цвет неба закрасит и небо, и землю.



Теперь выберите инструмент Заливка.

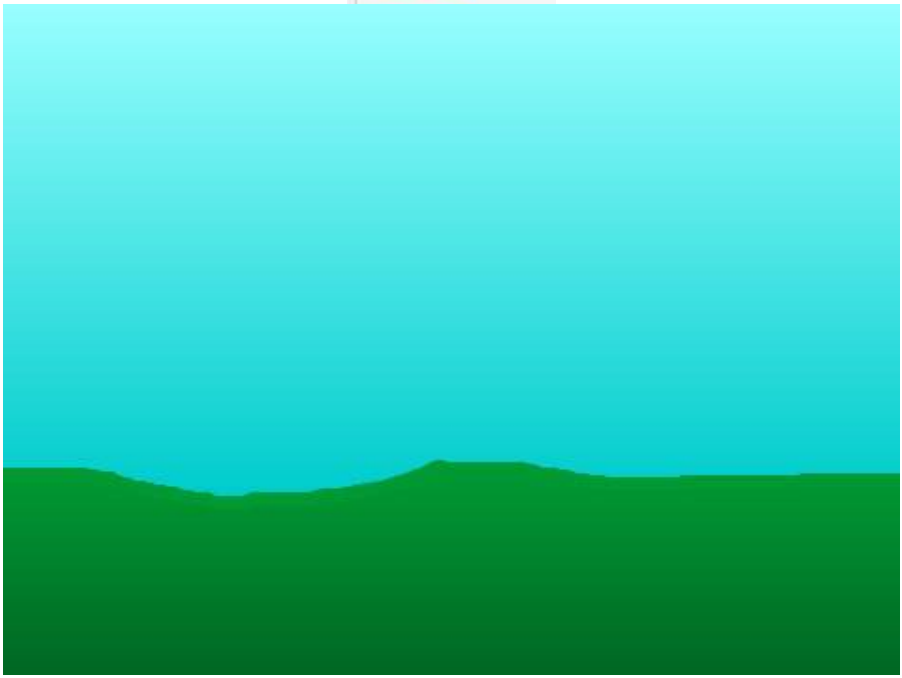
Заливка будет градиентная, поэтому выберите два цвета.



Выберите градиентный тип заливки.

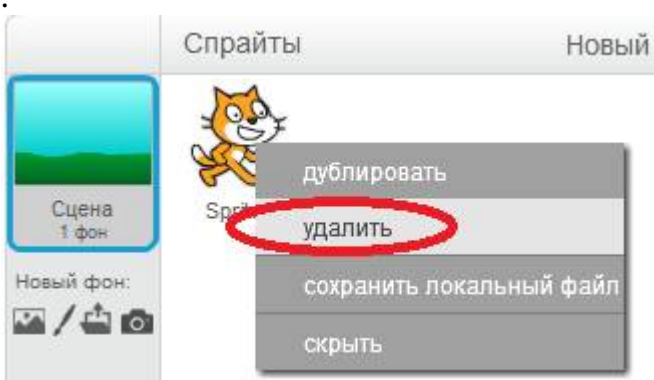
Залейте верхнюю часть рисунка небом.

Теперь выберите зелёные цвета, и залейте нижнюю половину рисунка травой.

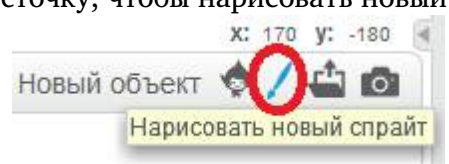


Получилось красиво!

Теперь нарисуем спрайты Базы, Зенитки, Самолета, Диверсанта, но сначала удалим Котика, он в этой битве не участвует.



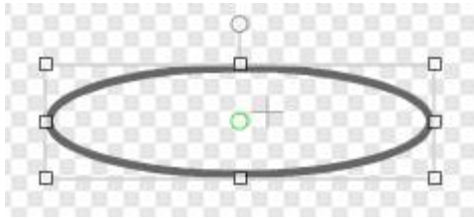

Кликните кисточку, чтобы нарисовать новый спрайт.




Измените масштаб.

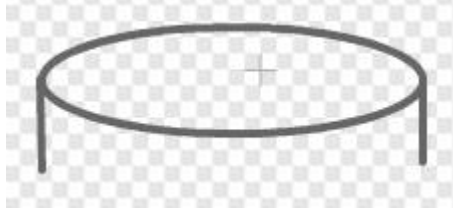
Выберите инструмент Эллипс.

Нарисуйте в центре небольшой овал (математическое название овала — эллипс).



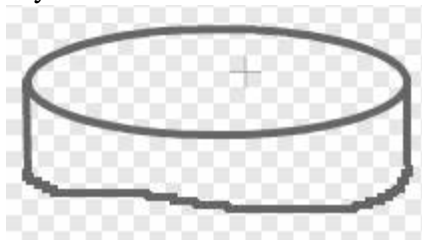
Выберите инструмент Линия.

Нарисуйте две вертикальные линии слева и справа.



Выберите инструмент Кисть.

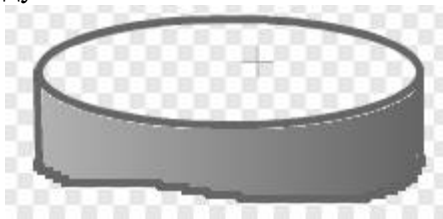
Нарисуйте внизу линию.



Выберите инструмент заливка.

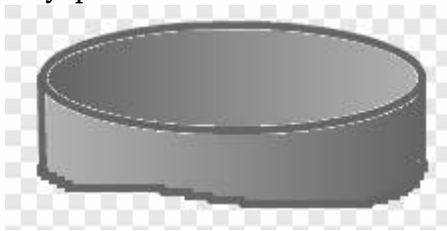
Выберите серые цвета.

Залейте ограду Базы.



Поменяйте цвета местами.

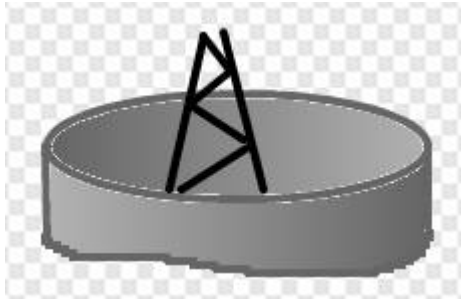
Залейте базу внутри.



Теперь нарисуем внутри постройку. Выберите Линию.

Выберите чёрный цвет.

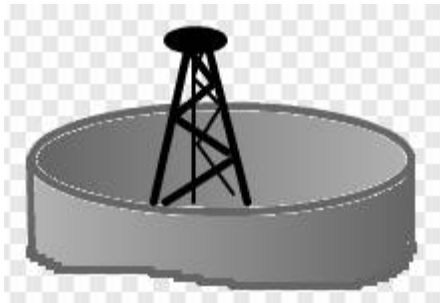
Нарисуйте вышку.



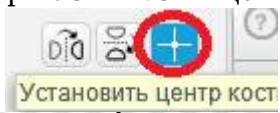
Наверху нарисуйте маленький черный овал. Выберите Эллипс.

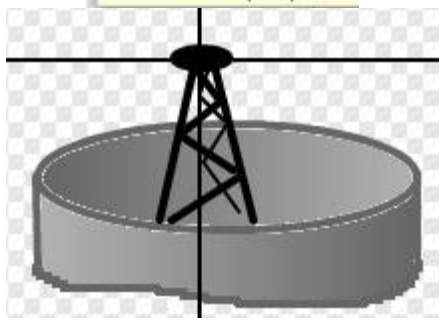
Выберите тип эллипса.

Нарисуйте маленький эллипс.



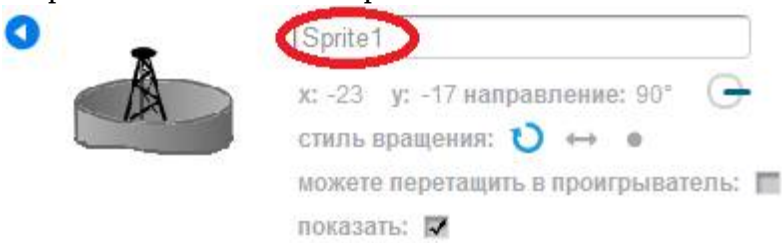
Установите точку привязки Базы в центр черного овала.



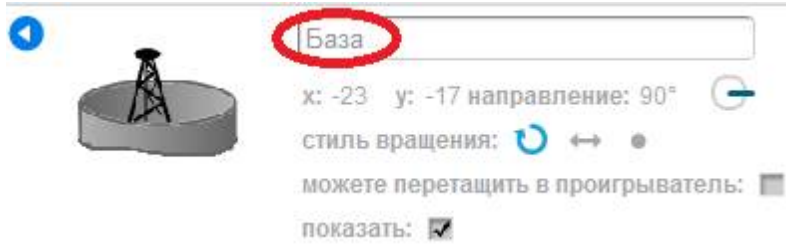



Переименуйте спрайт1 в Базу. Кликните на голубом кружочке.

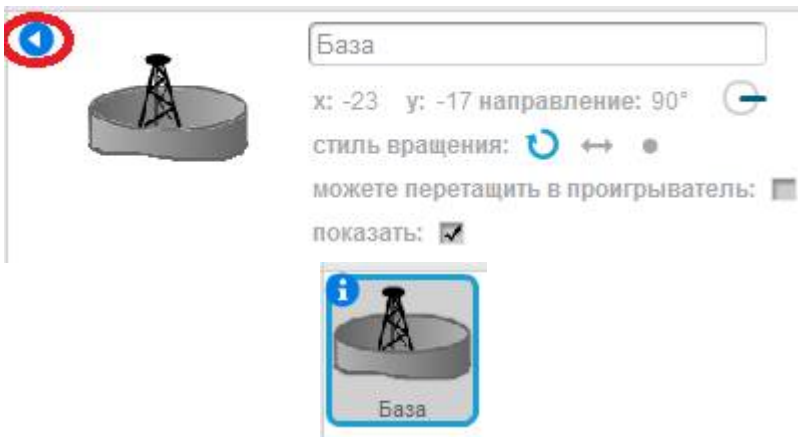
Откроется окно свойств спрайта.



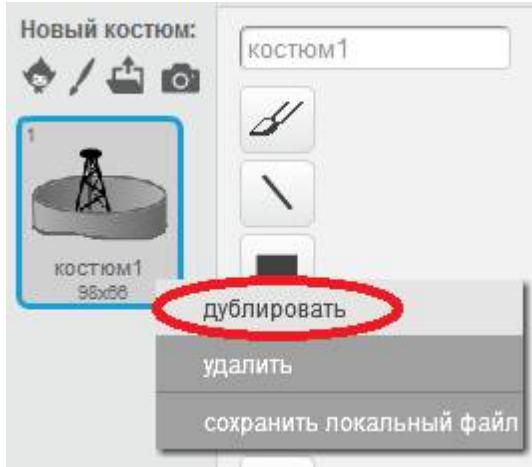

Измените имя на База.




Нажмите на голубой кружочек чтобы скрыть окно свойств спрайта.

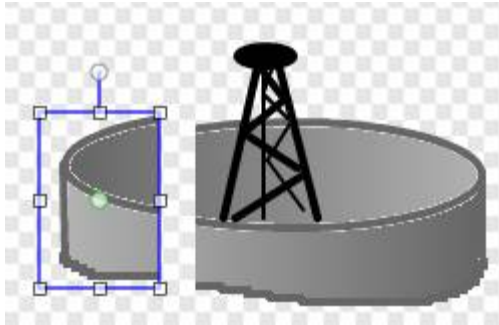



Теперь давайте сделаем Базе ещё девять костюмов для анимации её разрушения. Когда Диверсант доберётся до Базы — она, с огнем и дымом, взлетит на воздух, и превратится в груду обломков. Дублируйте первый костюм Базы.

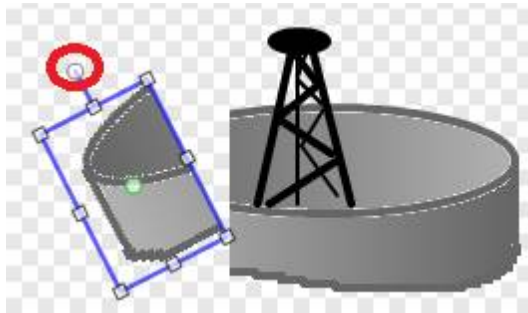



Кликните на инструмент Выбора.

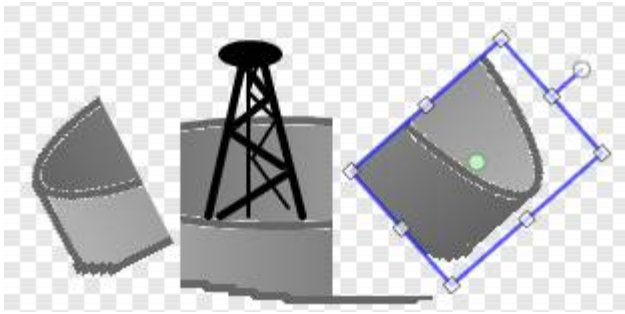
Выделите часть Базы и отодвиньте немного в сторону.



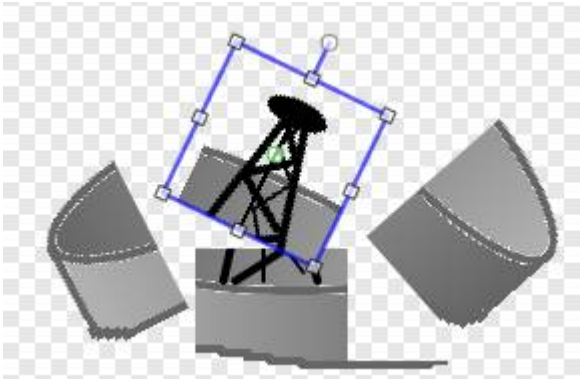
Теперь наведите курсор на верхний кружочек и поверните выделенную область.



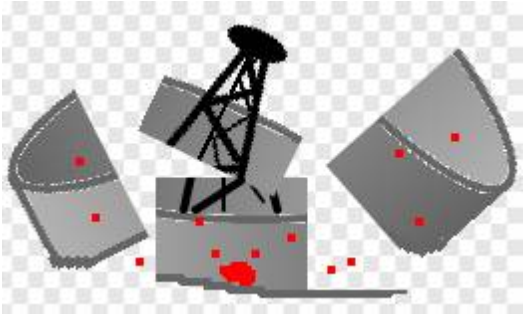
Выделите ещё часть Базы, отодвиньте её и поверните.



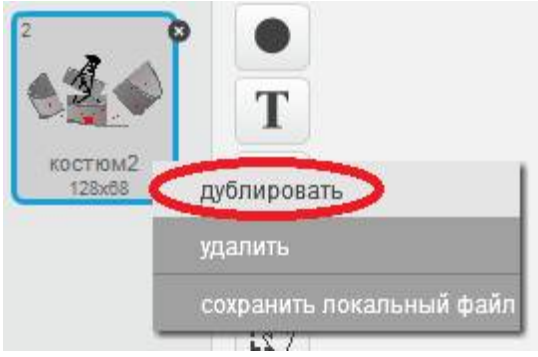
И ещё раз.



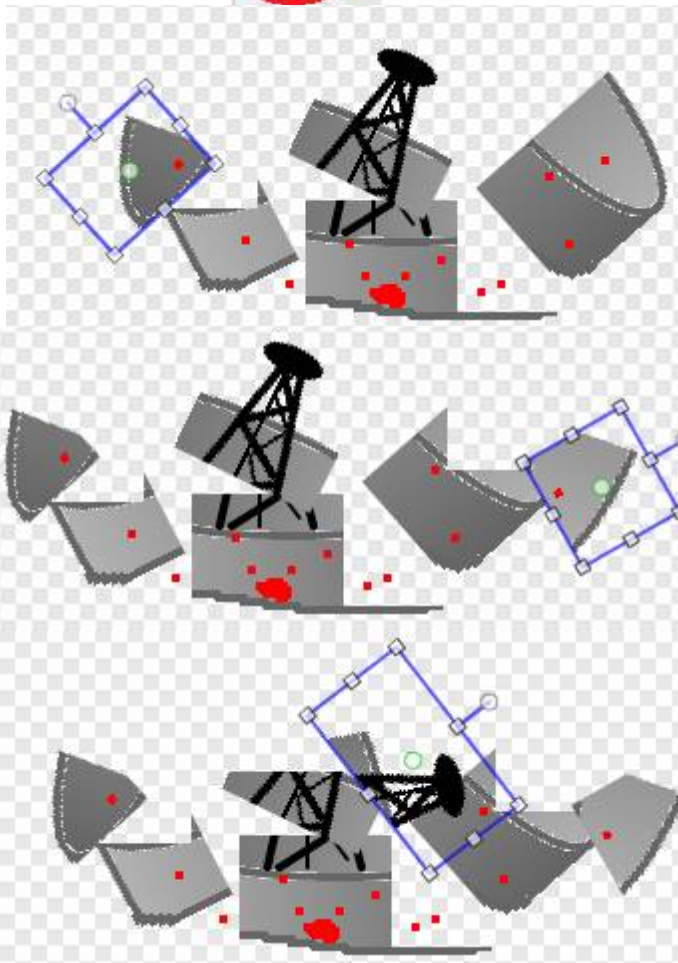
Сейчас выберите Кисть, и покройте осколки Базы красными точками. В центре нарисуйте вспышку взрыва.



Теперь дублируйте Костюм2.




Третий костюм также разделите на кусочки, ещё немного их раздвиньте и подрисуйте больше огня.



В результате должно получится что-то вроде этого.

Дублируйте Костюм3 и по аналогии с предыдущими костюмами превратите его в такой Костюм4.

Дублируйте Костюм4 и превратите его в такой Костюм5.

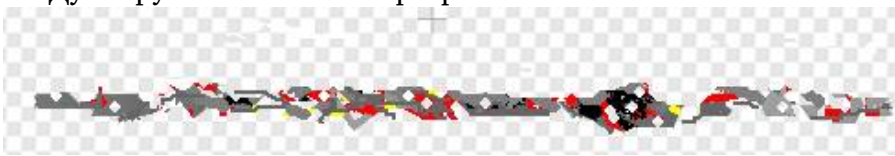
Дублируйте Костюм5 и превратите его в такой Костюм6.

Дублируйте Костюм6 и превратите его в такой Костюм7.

Дублируйте Костюм7 и превратите его в такой Костюм8. Не забывайте про ластик. Стирайте лишний дым и немного стирайте обломки, как будто они сгорели.

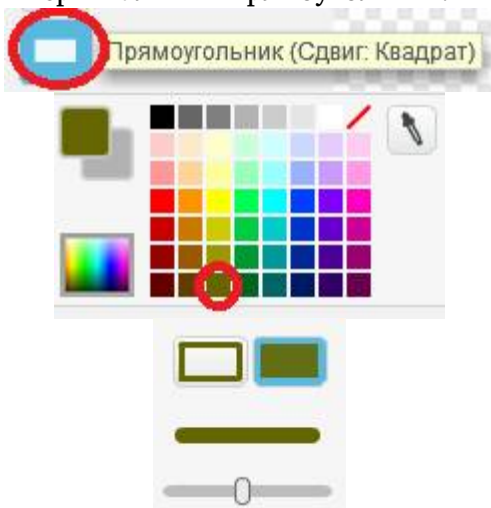
Дублируйте Костюм8 и превратите его в такой Костюм9.

Дублируйте Костюм9 и превратите его в такой Костюм10.

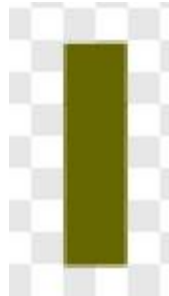


С Башней покончили. Теперь нарисуйте зенитку. Кликните на «Нарисовать новый спрайт».

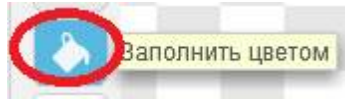
Зенитка выглядит очень просто. Это тоненький прямоугольник. Выберите инструмент Прямоугольник, и нарисуйте узкий вертикальный прямоугольник.




Увеличте масштаб. Нажмите два раза на плюсик.



Выберите инструмент Заливка.

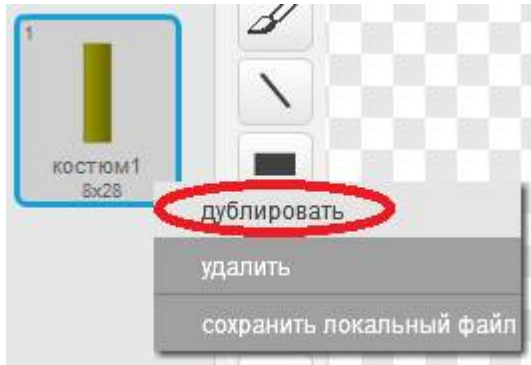



Выберите два близких цвета.

Вот такая будет у нас Зенитка, похожа на короткую трубу.

Задайте точку привязки на нижнем конце Зенитки. Точка привязки, это точка, вокруг которой будет вращаться Зенитка.

Нарисуйте ей второй костюм — как она стреляет. Сначала просто дублируйте костюм.

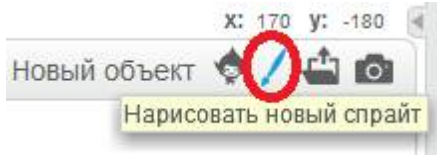



Подрисуйте на конце Зенитки пламя от выстрела. Выберите Кисть красного цвета.

Обратите внимание, точка привязки осталась внизу Зенитки. Переименуйте Зенитку.

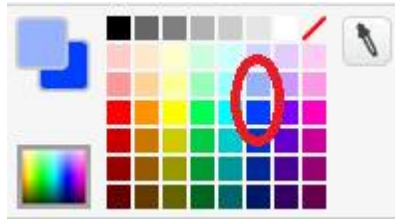
Теперь давайте нарисуем самолет. Конечно, можно взять готовый из папки с картинками, но лучше еще немного поучимся рисовать.

Нажмите «Нарисовать новый спрайт».



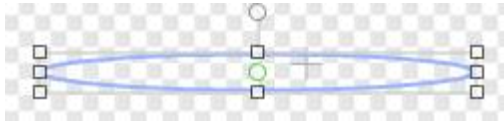

Выберите инструмент Эллипс.

Выберите голубой цвет.



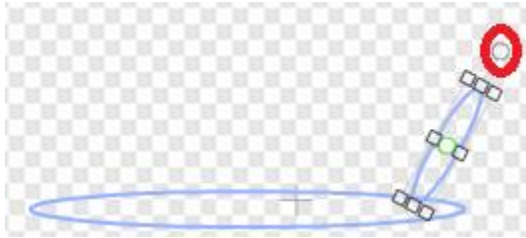
Линию сделайте потоньше.

Нарисуйте тоненький овальчик.



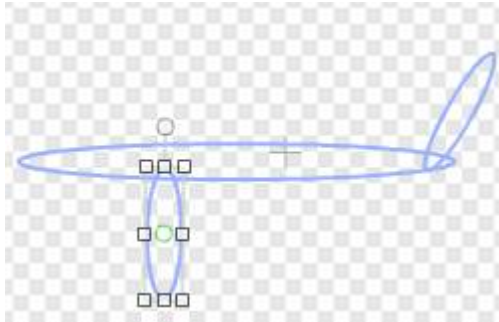
Нарисуйте второй овальчик.

Наведите курсор на верхний кружок, и поверните второй овальчик.



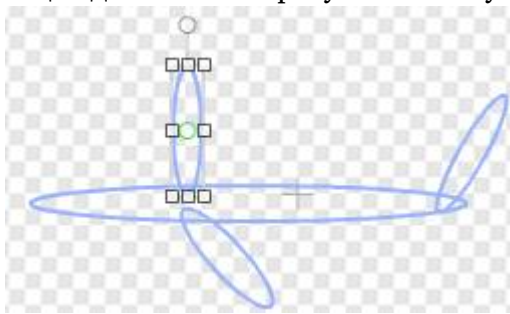
Нажмите на него мышкой, и подвиньте на край первого овала. Киль готов.

Теперь нарисуем крыло. Нарисуйте овал — левую половину крыла.

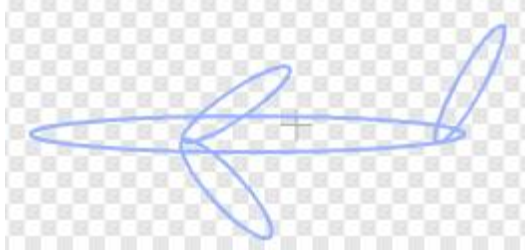


Поверните его за кружочек и подвиньте как надо.

Нарисуйте ещё один овал — правую половину крыла.

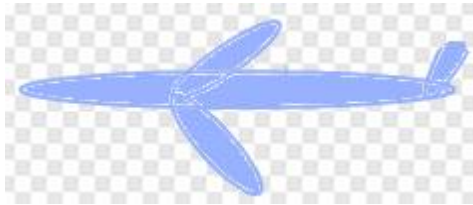


Поверните его за кружочек и подвиньте как надо.



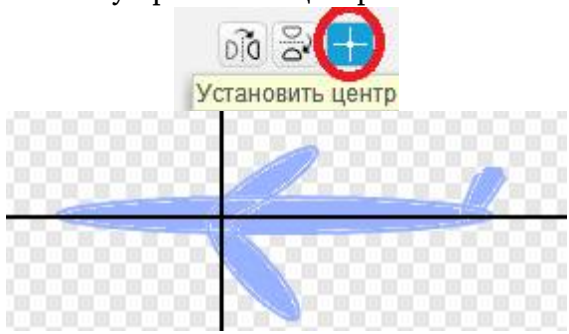
Теперь выберите заливку, и залейте его этим же голубым цветом.

Не очень красивый, зато свой! Надеюсь, у Вас самолетик получится красивее. Кажется, у меня киль великоват. Возьму ластик и подправлю.

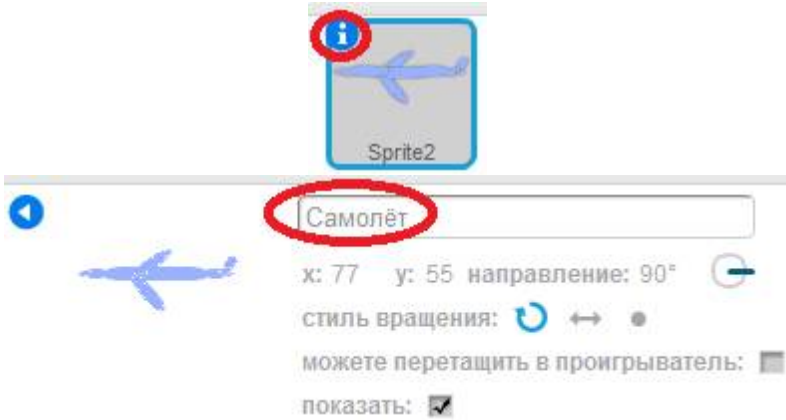


Вот так и оставим.

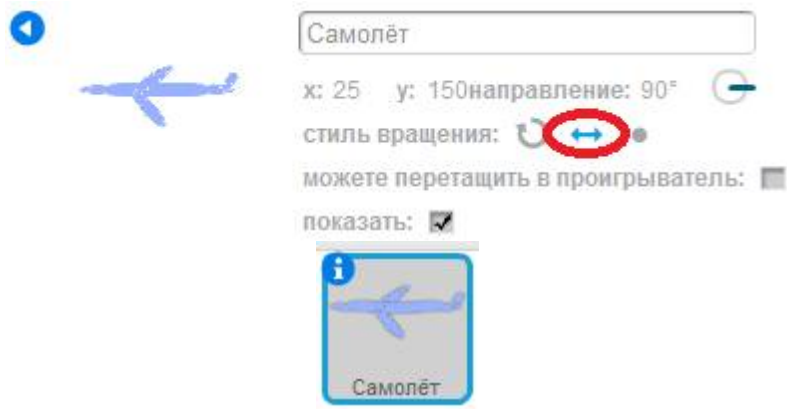
Установите точку привязки в центр Самолёта.




Не забудьте переименовать Самолёт.

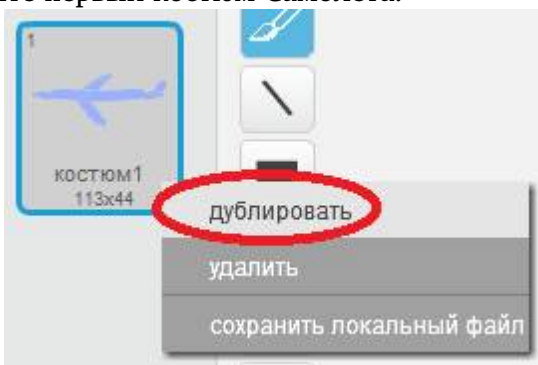



Не забудьте нажать на двойную стрелочку, чтобы Самолёт не мог летать вверх ногами.

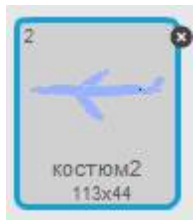



Теперь надо сделать Самолёту 5 костюмов разной степени дырявости. На втором костюме нарисуем одну дырочку, на третьем — две дырочки, на четвертом три дырочки и легкий дымок, на пятом 4 дырочки и сильный дым.

Дублируйте первый костюм Самолёта.




Выберите Костюм2.

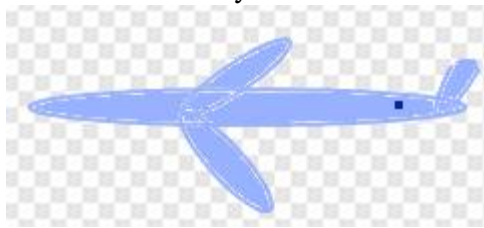



Возьмите Кисть.

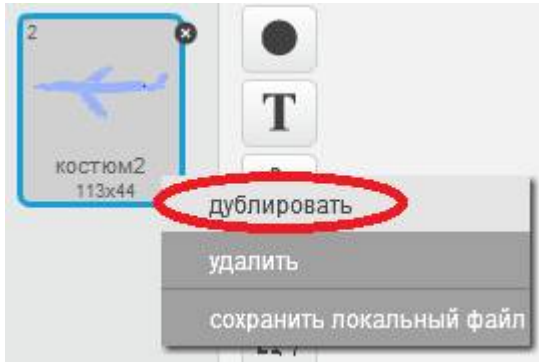
Выберите синий цвет.

Не очень тонкий размер Кисти.

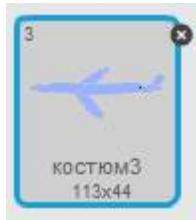
Поставьте на Самолёте точку.



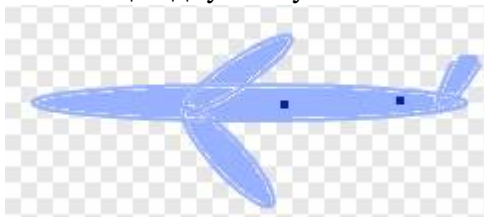
Дублируйте Костюм2.



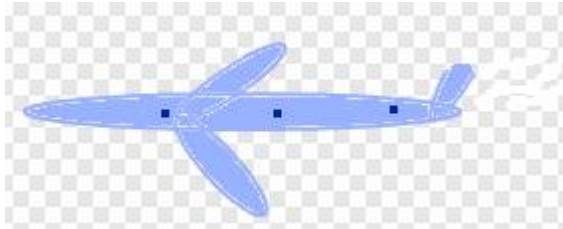

Выберите Костюм3.




Поставьте на нём ещё одну точку.



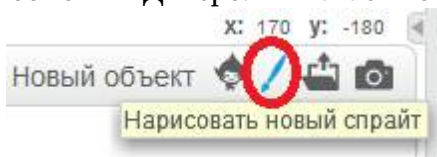
Дублируйте Костюм3, и поставьте ещё одну точку на Костюме4. Белым цветом подрисуйте маленький дымок.



Здесь дымок не очень хорошо видно, он будет хорошо смотреться на фоне неба.

Дублируйте Костюм4. На пятом костюме нарисуйте ещё одну дырку и добавьте густые клубы дыма.

Самолёт готов. Теперь нарисуем Диверсанта. У него будет три костюма. Первый костюм — Диверсант летит на парашюте, второй и третий костюм — Диверсант шагает по земле.



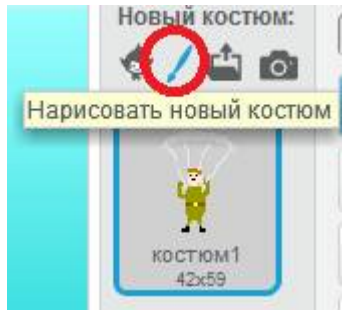

Возьмите кисть.

Два раза кликните на линзу с плюсиком — увеличьте масштаб до 800%.

Толщина Кисти не очень большая. Глаза и рот нарисуйте ещё более тонкой Кистью.

Нарисуйте Диверсанта с парашютом. У меня получился вот такой.

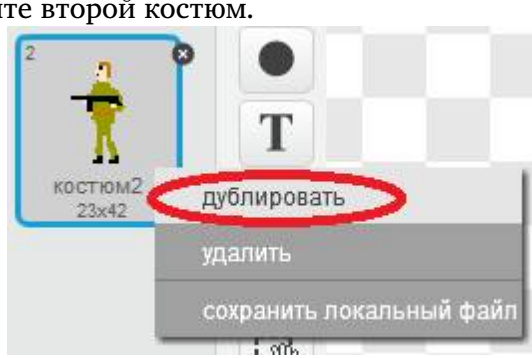
Нарисуйте второй костюм.




Рисуйте в таком же масштабе и такими же цветами.

Какой-то Иванушка дурачок получился. Постарайтесь нарисовать покрасивее.

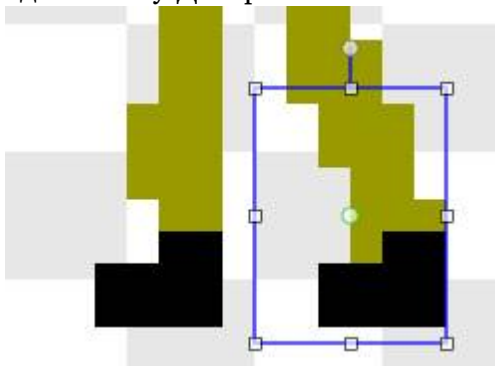
Скопируйте второй костюм.



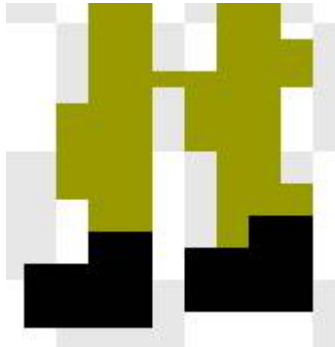

Увеличьте масштаб до 1600%.

Кликните инструмент Выбора.

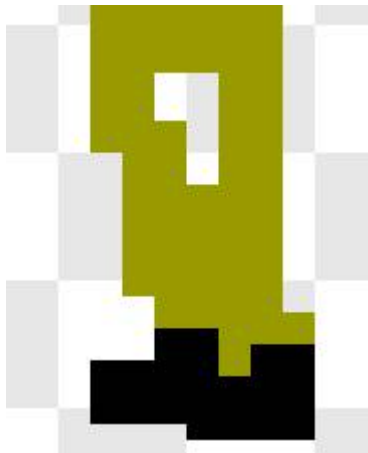
Выделите заднюю ногу Диверсанта.



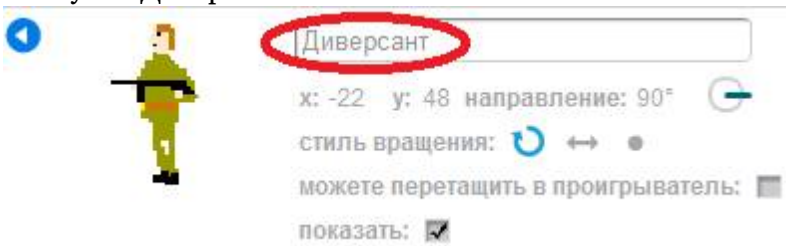
И передвиньте её влево.



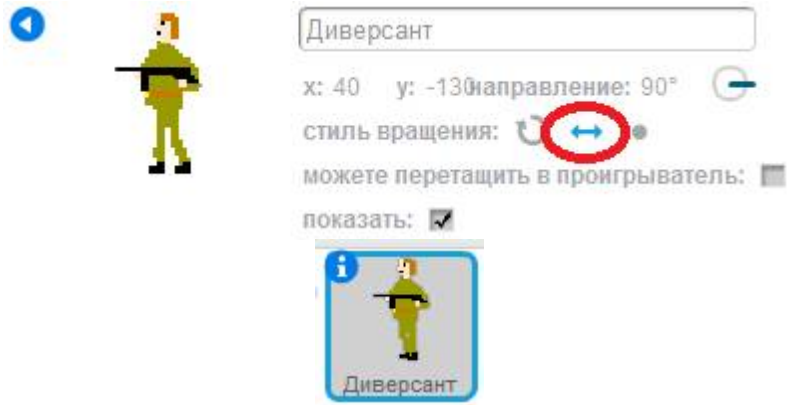
Сотрите ластиком лишнее. Так же подвиньте и переднюю ногу немного вправо и сотрите лишнее. Получится что-то вроде этого.



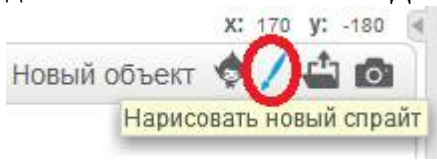
Пощёлкайте мышкой на втором и третьем костюмах по очереди, будет похоже, что Диверсант шагает. Сейчас переименуйте Диверсанта.




Не забудьте нажать на двойную стрелочку, чтобы Диверсант не мог летать вверх ногами.



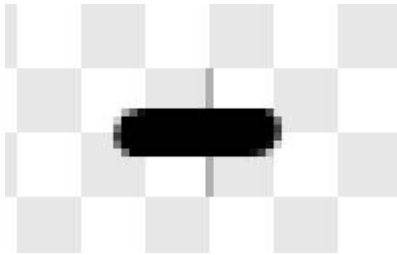

Теперь надо нарисовать Снаряд, который будет вылетать из Зенитки и взаимодействовать с Самолётом и Диверсантом.




Снаряд — это коротенькая черная линия.

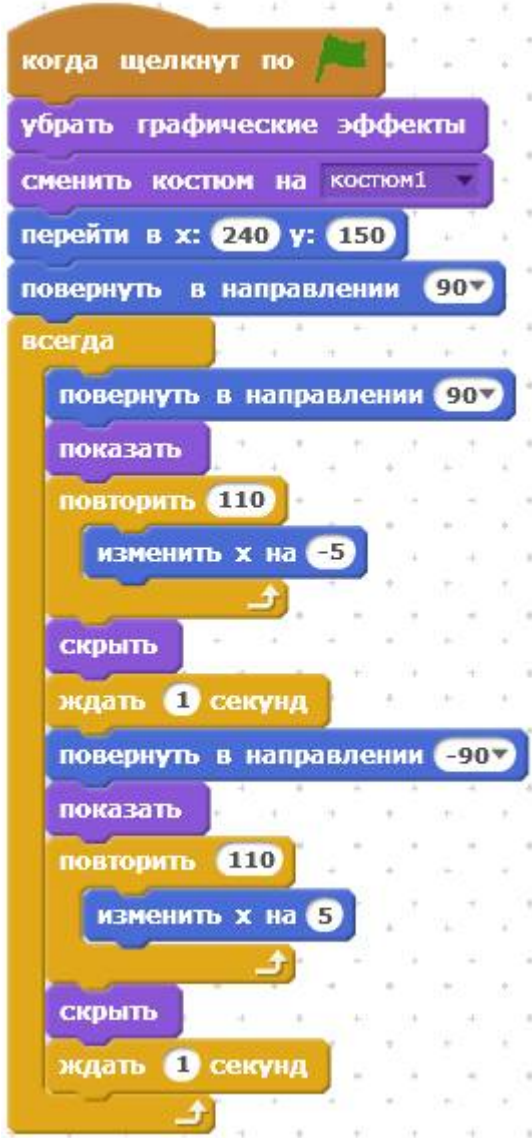
Укрупните масштаб. Дважды нажмите на плюсик.

Выберите черный цвет, и нарисуйте коротенькую линию. Рисуйте не самой тонкой линией.

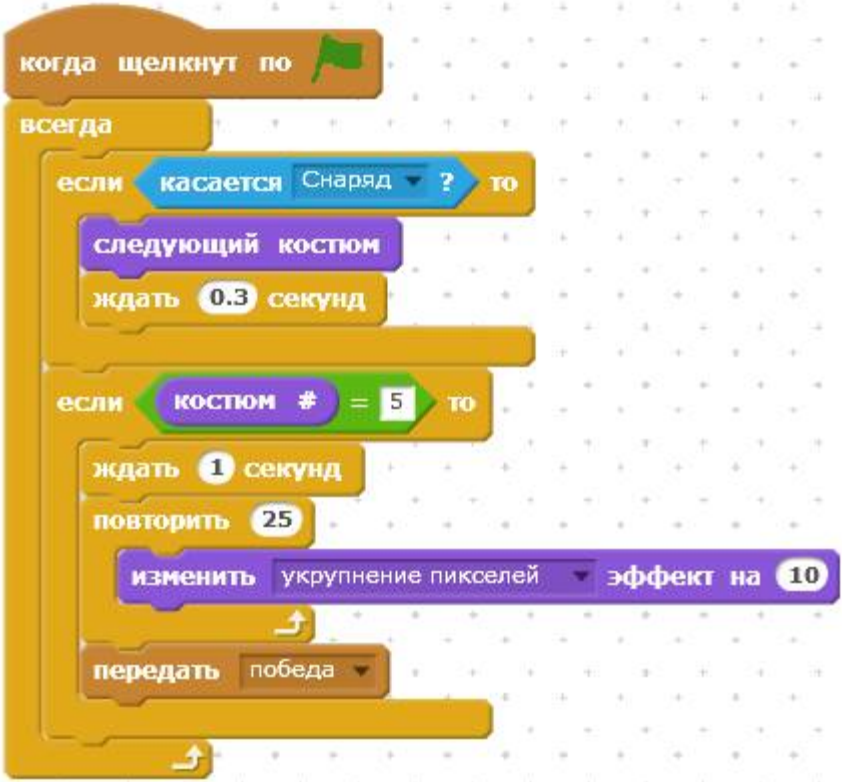


Переименуйте Снаряд.

Настало время запрограммировать поведение спрайтов. Начнём с Самолёта. Сделайте Самолёту два скрипта. Это первый скрипт Самолёта.

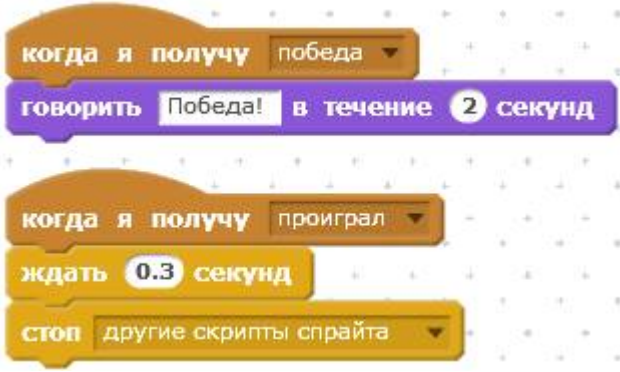



Это второй скрипт Самолёта.

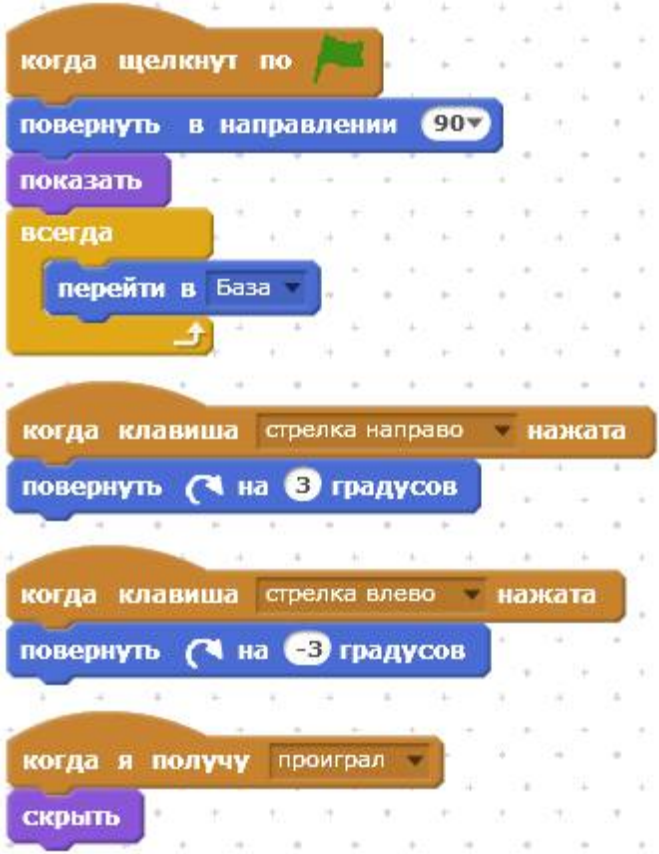



Теперь сделайте три скрипта для Базы. Это первый скрипт.

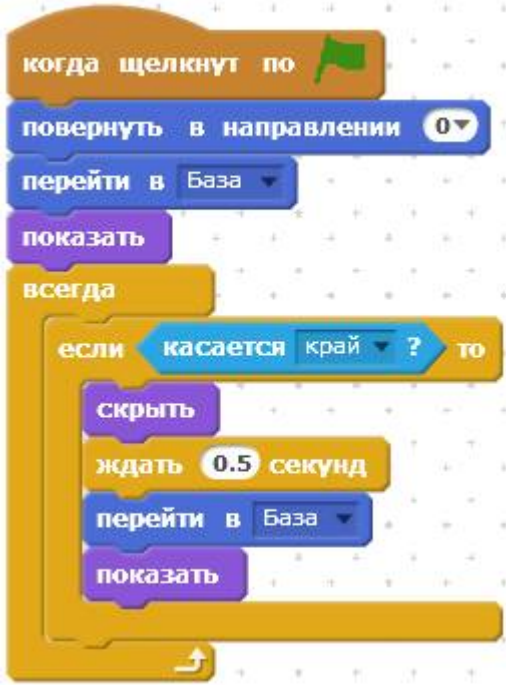
Ещё два скрипта.


когда я получу победа
говорить Победа! в течение 2 секунд
когда я получу проиграл
ждать 0.3 секунд
стоп другие скрипты спрайта

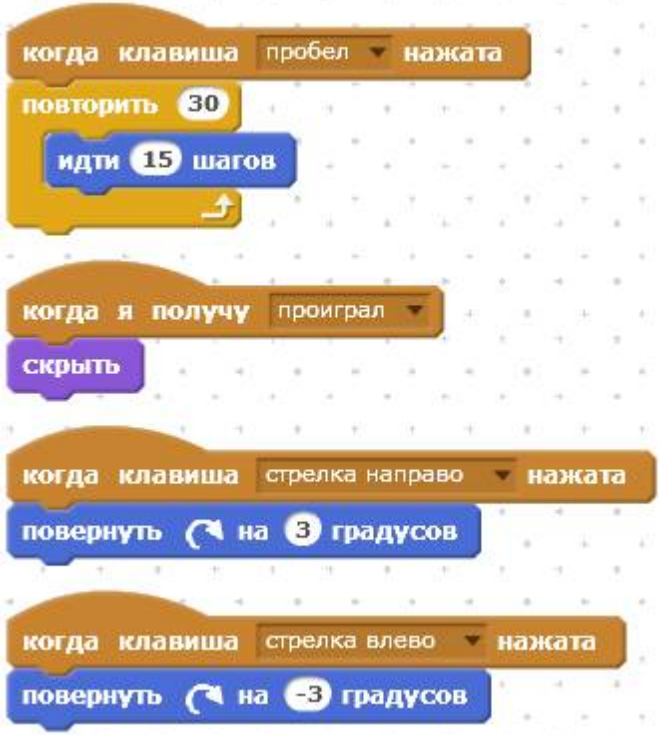
Сделайте четыре скрипта для Зенитки.



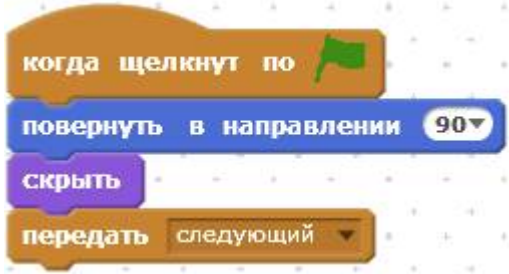

Ещё пять скриптов для Снаряда. Первый скрипт Снаряда.




И ещё четыре скрипта для снаряда.

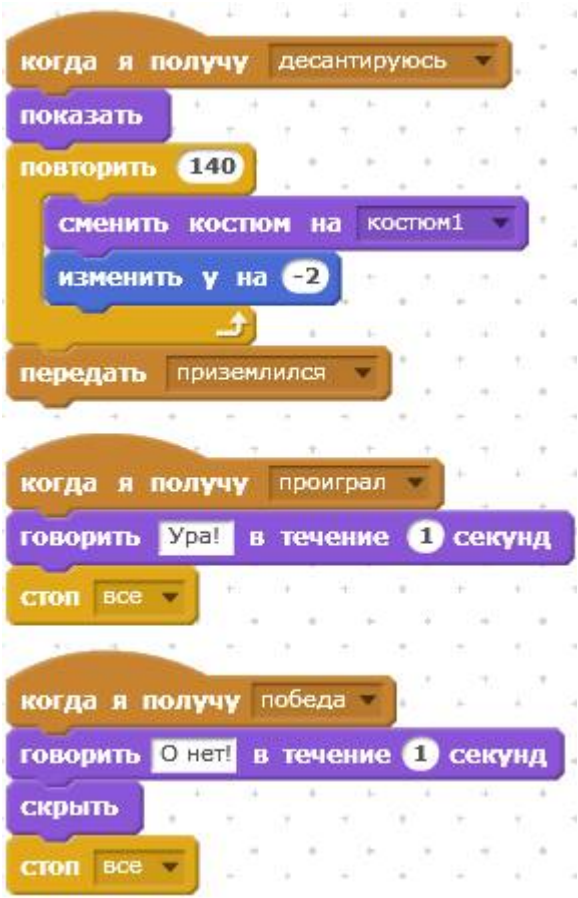



Теперь сделайте семь скриптов для Диверсанта. Первый скрипт.

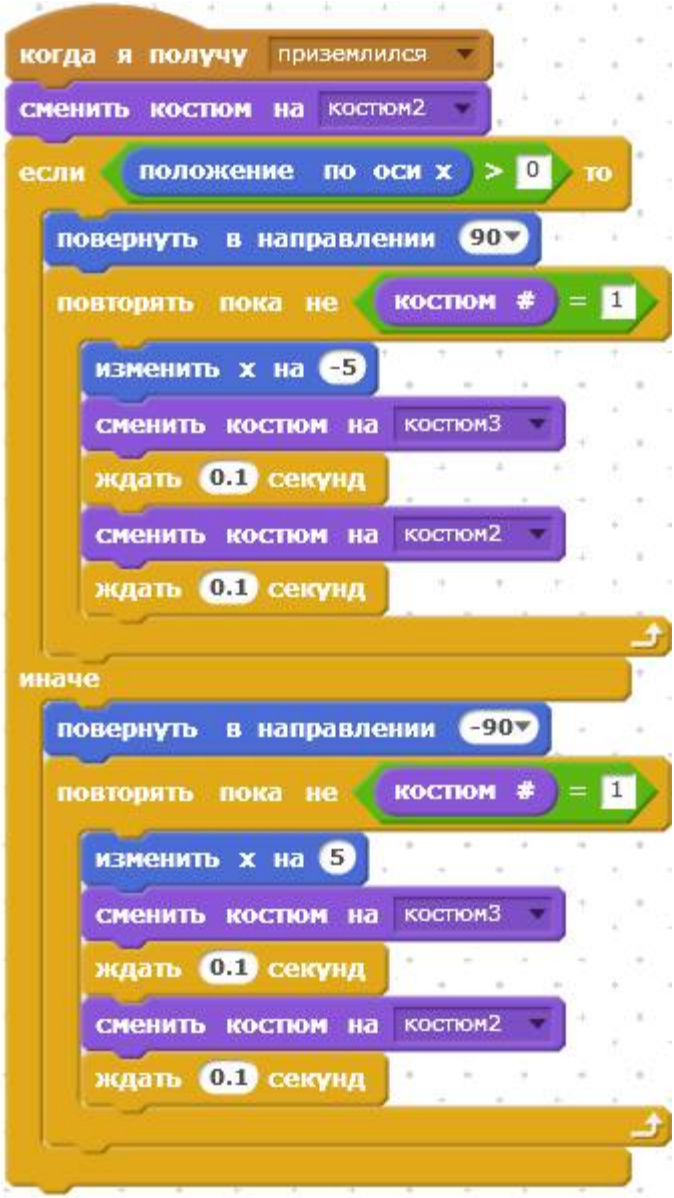



Ещё два скрипта для Диверсанта.

И ещё три скрипта для Диверсанта.




Седьмой скрипт для Диверсанта.


когда я получу приземлился
сменить костюм на костюм2
если положение по оси x > 0 то
повернуть в направлении 90
повторять пока не костюм # = 1
изменить x на -5
сменить костюм на костюм3
ждать 0.1 секунд
сменить костюм на костюм2
ждать 0.1 секунд
иначе
повернуть в направлении -90
повторять пока не костюм # = 1
изменить x на 5
сменить костюм на костюм3
ждать 0.1 секунд
сменить костюм на костюм2
ждать 0.1 секунд

Программа готова. Запустите и попробуйте поиграть.

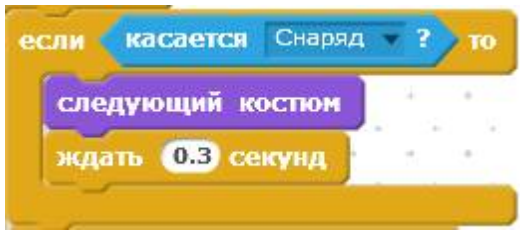
Объяснение работы программы.

Самолёт стартует из правого верхнего угла экрана.

Потом поворачивается носом влево и плавно летит — 110 раз по 5 точек смещается влево. Залетает за край экрана, пропадает и ждёт 1 секунду.

Теперь Самолёт появляется уже с левой стороны. Направление изменено на противоположное и смещается он 110 раз по 5 точек вправо. Залетает за край экрана, пропадает и ждёт 1 секунду.

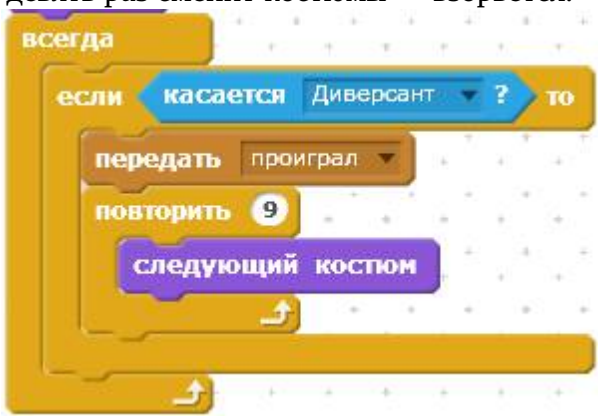
Если Самолёт коснётся снаряда, то он сменит костюм на более дырявый.




Если в Самолёт попали несколько раз и номер костюма Самолёта равен пяти, то цель игры достигнута. Самолёт подбит. Он быстренько исчезает с эффектом укрупнения пикселей и передаёт событие «победа».

База всегда стоит на одном месте. В точке (0 ; -150).

База стоит на месте и ничего не делает, ждёт Диверсанта. Когда он коснётся Базы, она передаст событие «проиграл» и быстренько девять раз сменит костюмы — взорвётся.




Зенитка тоже всегда стоит на месте в точке привязки Базы — там, где у Базы задан центр костюма.

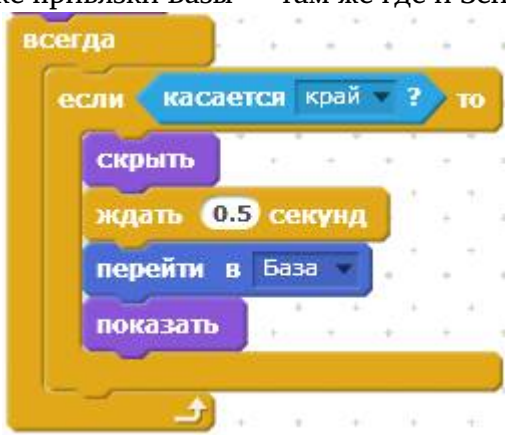
Зениткой можно управлять с помощью стрелок. Если числа 3 и -3 заменить на меньшие, то Зенитка будет поворачиваться медленнее и играть будет труднее.

когда клавиша стрелка направо нажата
повернуть на 3 градусов
когда клавиша стрелка влево нажата
повернуть на -3 градусов

Снаряд в начале игры находится там же где и Зенитка — в центре костюма Базы.

Если, нажав на пробел, произвести выстрел, то Снаряд будет двигаться вперёд 30 раз по 15 шагов.

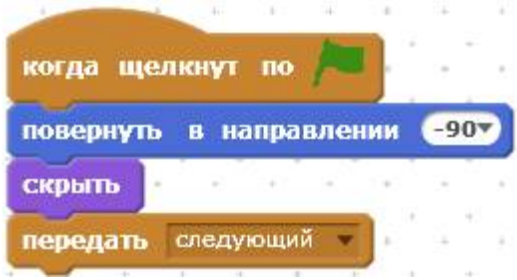
Если во время полёта снаряд долетит до края экрана, то он скроется, подождёт половину секунды и снова мгновенно окажется в точке привязки Базы — там же где и Зенитка.



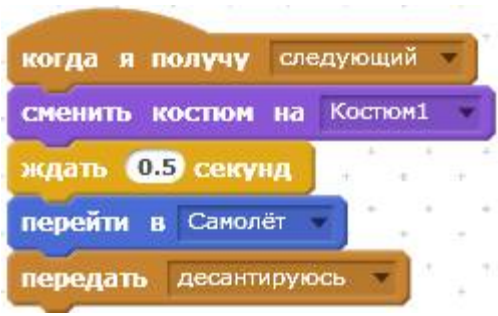

Направление Снаряда изменяется вместе с изменением направления Зенитки, тоже стрелками вправо и влево. Если Вы пожелаете усложнить игру и измените числа 3 и -3 у Зенитки, не забудьте изменить их и у Снаряда.

когда клавиша стрелка направо нажата
повернуть на 3 градусов
когда клавиша стрелка влево нажата
повернуть на -3 градусов

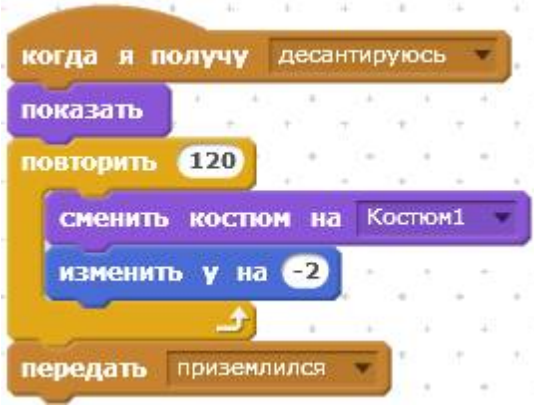
Программа Диверсанта самая сложная, она состоит из нескольких этапов. Первый этап — полёт на Самолёте, потом полёт на парашюте, третий этап — движение к Базе по земле. Вначале Диверсант невидим, он передаёт сам для себя событие «следующий». Это означает начало первого этапа — полёта на Самолёте.




Когда он получит это событие, то начинается подготовка к прыжку. Он по-прежнему невидим. Диверсант одевает костюм с парашютом, половину секунды набирается смелости, идет в точку привязки Самолёта (в центр Самолёта), и передаёт событие «десантируюсь».

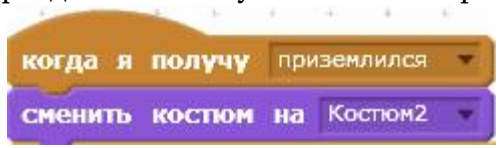



Когда он получит событие «десантируюсь», то появится в костюме парашютиста и 120 раз сдвинется вниз на 2 точки — опустится на землю. Когда приземлится, то передаст событие «приземлился».

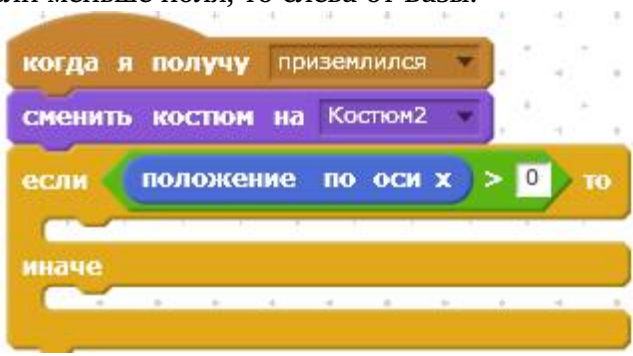



Дальше начинается самый сложный скрипт. Он состоит из двух симметричных частей. Одна часть работает если он приземлился слева от Базы, он поворачивается лицом направо и идёт направо. Другая часть скрипта работает если он приземлился справа от Базы, то он поворачивается лицом налево и двигается налево. Все прогулки по земле происходят в костюме2 и костюме3.

Сначала перейдем к костюму2 — снимем парашют.




Потом проверяем где приземлился Диверсант — слева или справа от оси X. Если положение по оси X больше ноля, то справа от Базы, если меньше ноля, то слева от Базы.



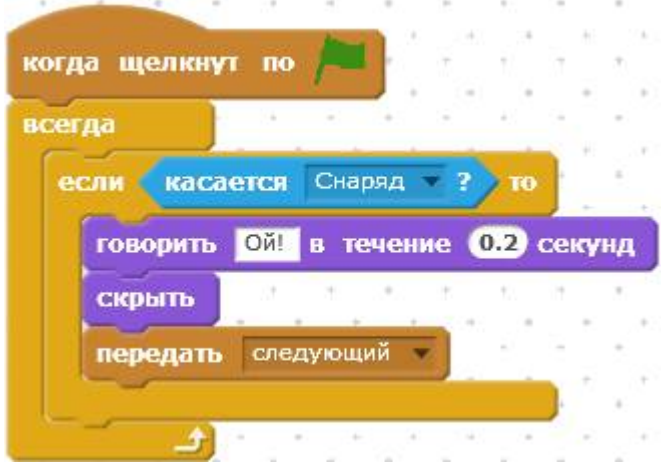

Если он приземлился справа, то он изменяет X на -5 (двигается влево) пока не оденет парашют (костюм = 1, а это костюм с парашютом). Всё время, пока он не оденет парашют, он двигается влево, постоянно изменяя костюмы со второго на третий. Это анимация ходьбы.

повернуть в направлении 90
повторять пока не костюм # = 1
изменить x на -5
сменить костюм на Костюм3
ждать 0.1 секунд
сменить костюм на Костюм2
ждать 0.1 секунд

Если он приземлился слева, то всё происходит симметрично. Он изменяет X на 5 (двигается вправо) пока не оденет парашют. Он двигается вправо, постоянно изменяя костюмы со второго на третий.

повернуть в направлении -90
повторять пока не костюм # = 1
изменить x на 5
сменить костюм на Костюм3
ждать 0.1 секунд
сменить костюм на Костюм2
ждать 0.1 секунд

Самый важный скрипт, от которого зависит жизнь Диверсанта — встреча со Снарядом. Если он встретится со снарядом, то скажет «Ой», передаст событие «следующий» и исчезнет. Этому Диверсанту не удалось уничтожить Базу, следующий будет пытаться сделать это.




Игра готова. Попытайтесь продержаться подольше.

Вопросы по игре «Диверсант».

1. Как изменить скорость ходьбы Диверсанта?
2. Как изменить скорость полёта Самолёта?
3. Как увеличить скорость полёта снаряда?
4. Управляется ли снаряд стрелками во время полёта?
5. В каком случае Диверсант, в которого попал снаряд, может взорвать Базу?

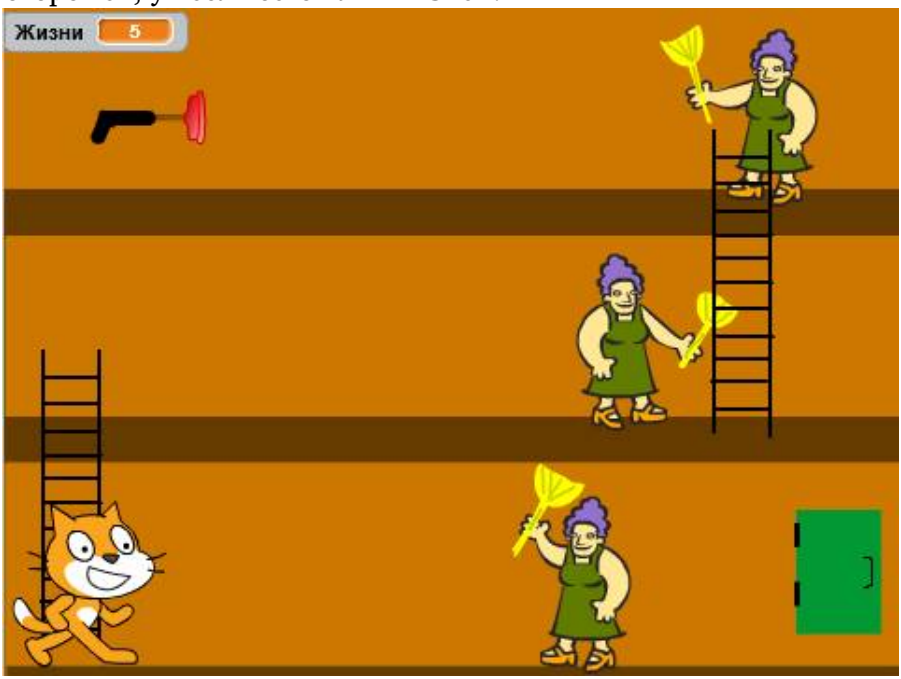
Задания к игре «Диверсант».

1. Сделайте так, чтобы Самолёт бросал бомбы.
2. Сделайте второй Самолёт, с которого также будут прыгать Диверсанты.
3. Сделайте разрывные снаряды. При подъеме на высоту Самолёта, такие снаряды взрываются, и поражают Самолёт осколками. Такими снарядами легче попасть в цель.

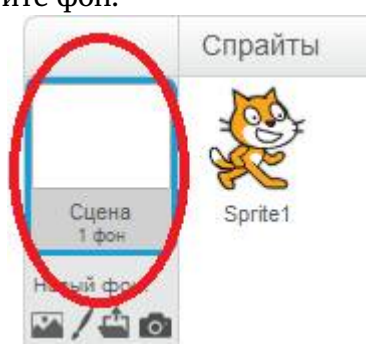
# ГЛАВА 8. БИТВА ЗА ДЕНЬ РОЖДЕНИЯ КОТА.

У Кота сегодня день рождения! Ему прислали подарок, а получить он его не может. Дорогу преградили злые Тётки с вениками. Они хотят выгнать кота из подъезда. Что же делать? На

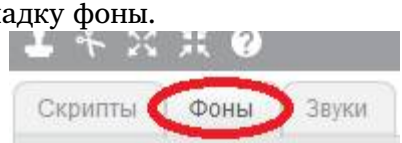
помощь придет секретное оружие — пистолет, стреляющий вантузами! Вперёд, Котик, подарок на четвёртом уровне, но будь осторожен, у тебя всего пять жизней!



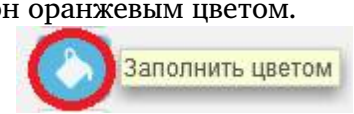

Сначала нарисуйте фон.

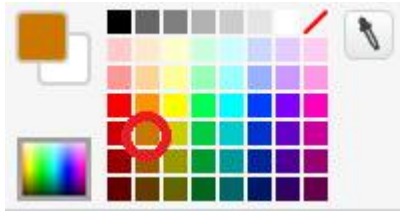



Выберите закладку фоны.

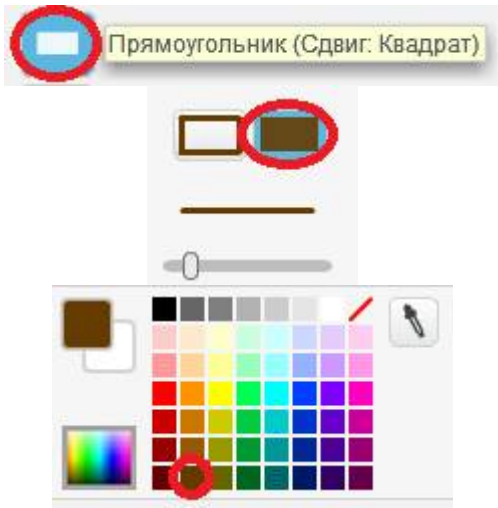



Залейте весь фон оранжевым цветом.



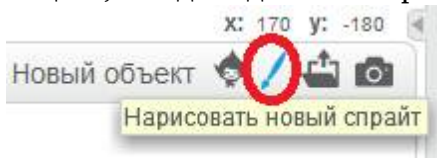


Потом нарисуйте три узких коричневых прямоугольника — это будут этажи.



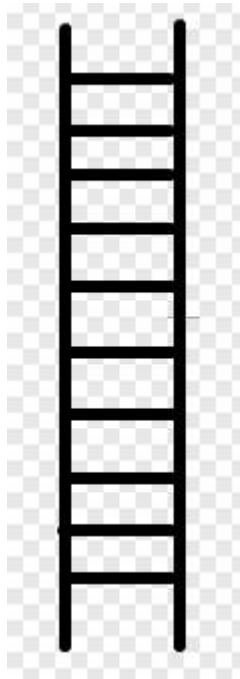

Старайтесь разделить этажами сцену ровно на три равные части. Нижний прямоугольник нарисуйте совсем тоненьким. Должно получиться примерно, как на картинке.

Теперь надо нарисовать две лестницы между этажами. Нарисуем их не на сцене, а виде отдельных спрайтов.

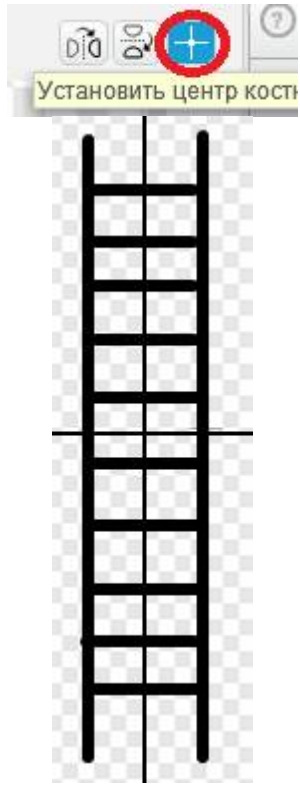



Выберите инструмент Линия.

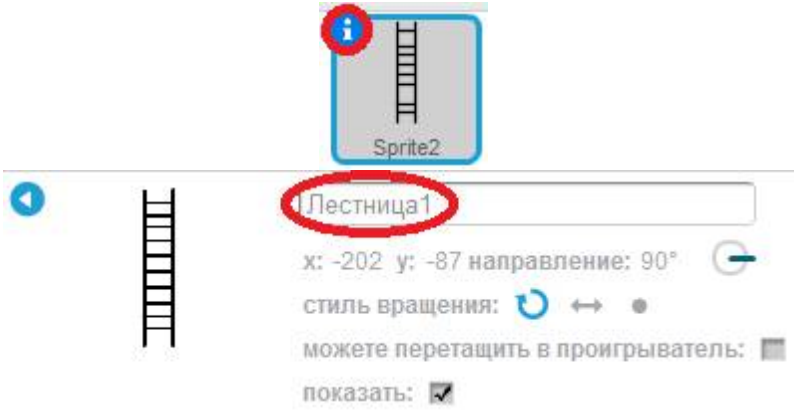
Рисуя лестницу, постоянно удерживайте нажатой клавишу Shift на клавиатуре. Это позволит нарисовать очень ровную лестницу, так как при нажатии этой клавиши все линии получаются только вертикальными или горизонтальными.



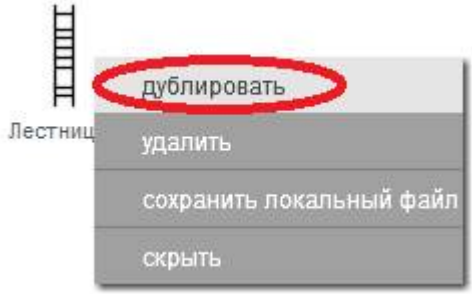
Обязательно задайте точку привязки в центре лестницы.



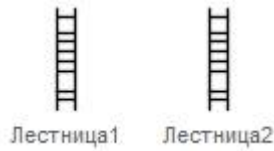

Переименуйте лестницу в Лестницу1.



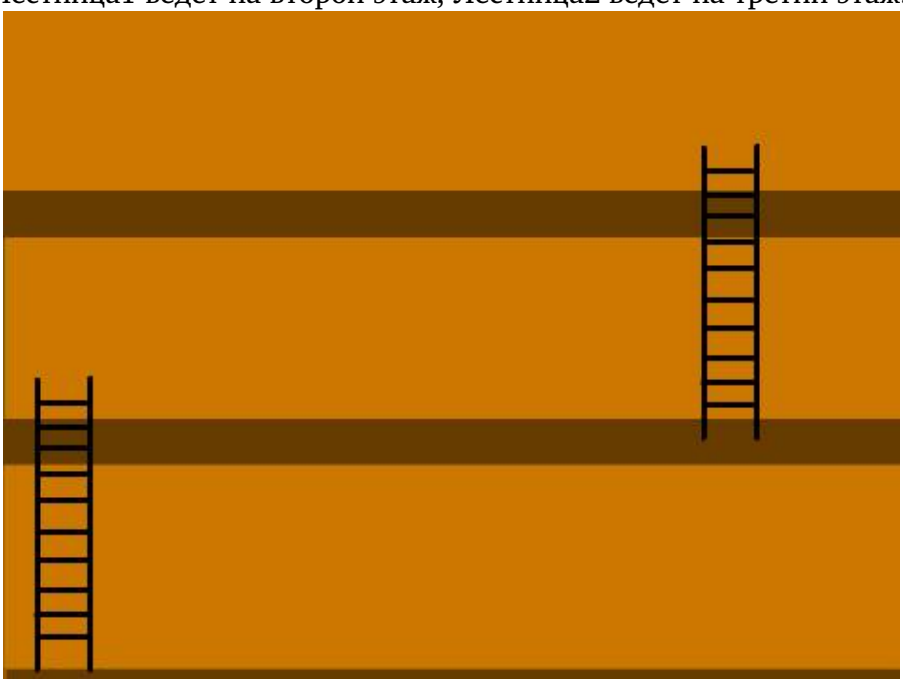

Нажмите на Лестницу1 правой кнопкой мышки и выберите «дублировать».



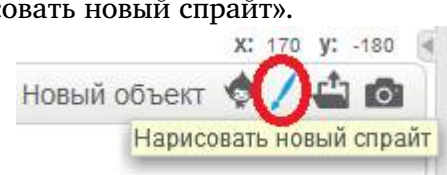

Теперь у нас есть две одинаковые лестницы. Кот сможет лазить по этажам!



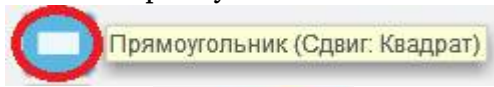

Разместите лестницы так, как показано на картинке. Лестница1 ведёт на второй этаж, Лестница2 ведёт на третий этаж.

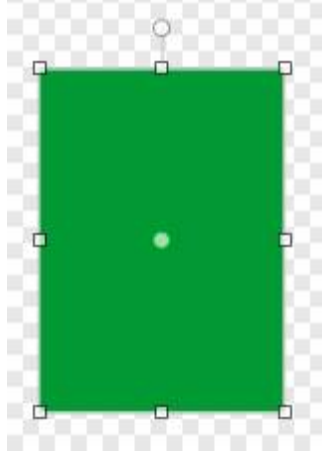


Теперь надо нарисовать три двери. Первая дверь нужна для перехода с первого уровня на второй, вторая ведёт на третий уровень, а через третью можно попасть на четвёртый уровень. Нажмите «Нарисовать новый спрайт».



Нарисуйте зелёный прямоугольник.

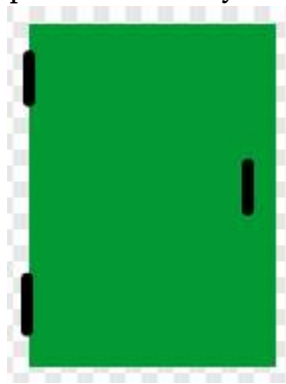




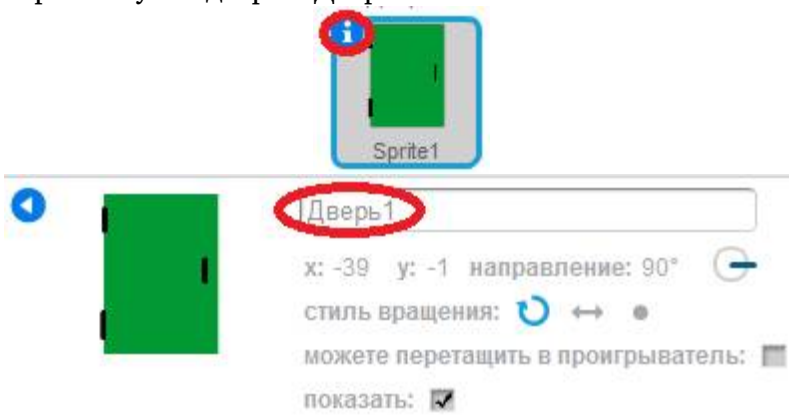
Чёрными линиями нарисуйте петли и ручку. Для того, чтобы рисовать точнее, увеличьте масштаб.

Выберите инструмент Линия.

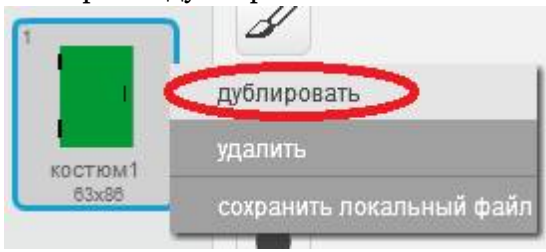
Рисуйте линии удерживая клавишу Shift.



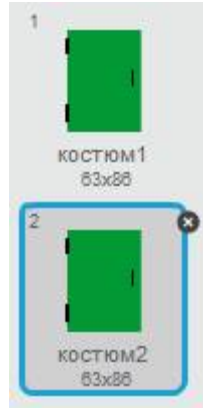
Переименуйте дверь в Дверь1.




Двери в нашей игре будут немного повреждаться при попадании Вантуза. Всего у каждой двери будет пять костюмов. Каждый следующий с большим количеством повреждений, чем предыдущий. Давайте сделаем костюм2. Нажмите правой кнопкой на костюм1, и выберите «дублировать».

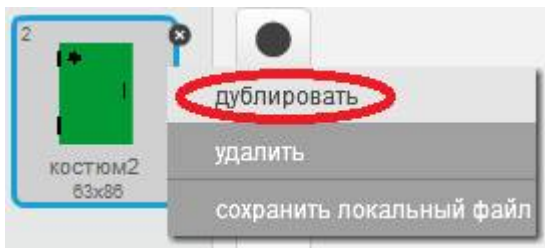


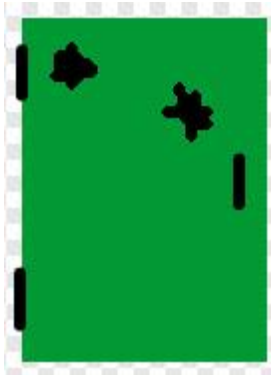

Костюм2 ничем не отличается от костюма1.



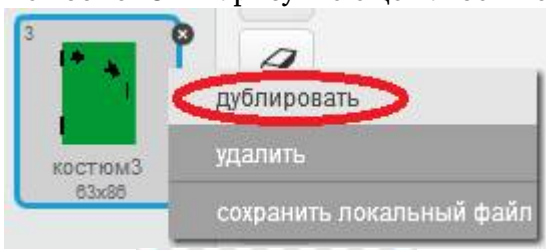

Давайте добавим ему одно повреждение. Выберите Кисть чёрного цвета и нарисуйте черное пятно.

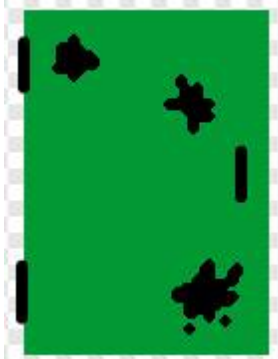
Теперь дублируйте костюм2, и подрисуйте ещё одно повреждение.



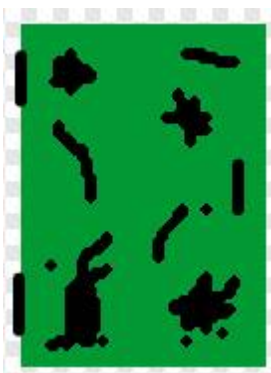


Дублируйте костюм3 и нарисуйте ещё какое-нибудь пятно.



Подобным образом создайте костюм5, он должен быть самым страшным.

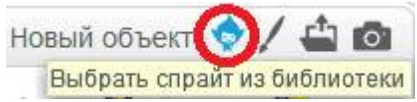


Программу для Двери1 сделаем потом, когда добавим все необходимые спрайты. А сейчас надо добавить вражеских

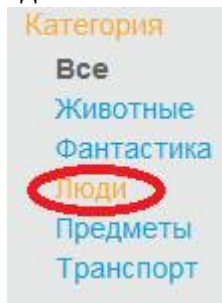
персонажей — вредных Тёть. За образец возьмём спрайт из библиотеки «Woman Waving» - «Тётя с завивкой на голове».

Woman Waving

Нам будет нужно добавить ей в руку веник, и сделать три костюма для анимации. Нажмите на кнопку «Выбрать спрайт из библиотеки».



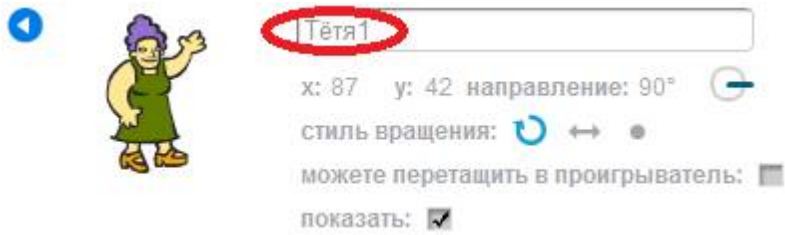

Выберите фильтр «Люди».




Добавьте нашу красавицу и нажмите внизу Ок.

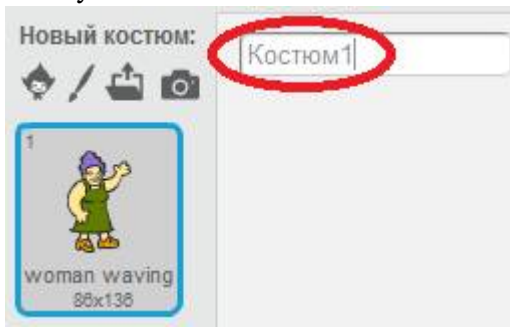
Woman Waving

Сразу переименуйте её в Тётю1.

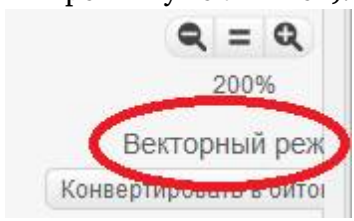



Откройте редактор изображений.

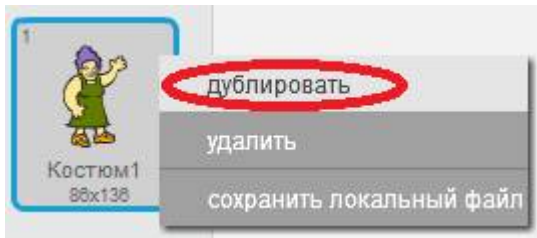
Сразу переименуйте костюм Тёти.




Обратите внимание, наша красавица нарисована в векторном формате, линии гладкие, отдельных пикселей (точек) не видно. Когда мы загружаем векторное изображение в Scratch, то редактор изображений автоматически переходит в векторный режим. Посмотрите в правый нижний угол редактора (нажимать туда не надо, векторный режим уже активен).




Сначала подготовим три костюма Тёти. На первом рука поднята вверх, на втором немного опущена, на третьем опущена ещё сильнее. Веник подрисуем потом. Дублируйте первый костюм Тёти.




Увеличьте масштаб.

Выберите инструмент «Изменение формы».

Кликните на черную обводку поднятой руки.

Появились опорные точки. Теперь нажмите на клавиатуре Ctrl-C. Кажется, ничего не произошло, но, на самом деле, выделенная рука скопировалась в память компьютера. Теперь нажмите Ctrl-V, вот и она, двигается в след за мышкой. Кликните на свободное место.

Рука скопирована. Теперь надо немного повернуть её. Нажмите на точку вращения и поверните копию руки.

Должно получиться вот так.

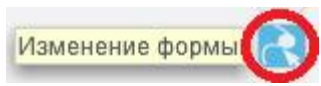
Теперь надо заменить руку. Снова выделите руку Тёти инструментом «Изменение формы».

Нажмите на клавиатуре кнопку Delete. Черная обводка руки будет удалена.

Перемещаем новую руку на место удалённой. Нажмите на инструмент Выбора.

Теперь нажмите на руку и тащите её на место.

Получилось немного не ровно. Сейчас поправим. Нажмите «Изменение формы».

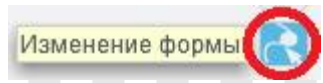



Нажмите на опорную точку и тащите её к плечу. Потом соседнюю.

Обводка немного толстовата. Переместите ещё одну точку.

Вот теперь нормально.

Следующая задача — переместить заливку внутрь контура. Выбранным инструментом кликните на заливку.

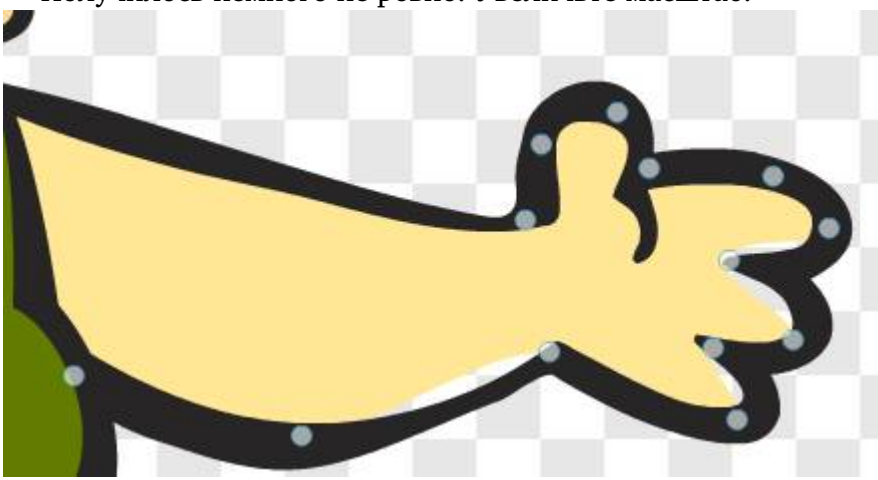


Начинаем сложную операцию. Опорные точки поочерёдно нужно переместить в места изгибов чёрной обводки. Начинаем

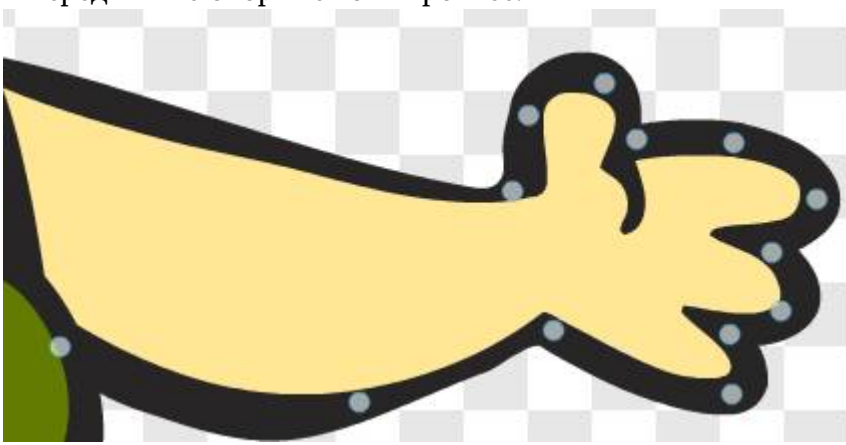
снизу. Очень важно передвигать точки поочередно. Если собьётесь, всегда можете нажать на стрелку «Отмена последнего действия».

Смотрите на рисунки и повторяйте.

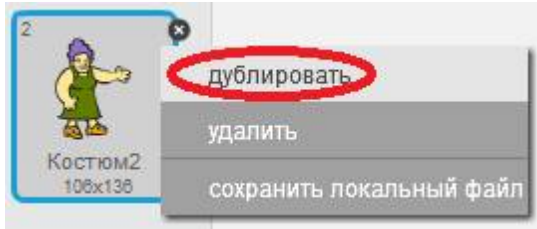
Получилось немного не ровно. Увеличьте масштаб.



Передвиньте опорные точки ровнее.

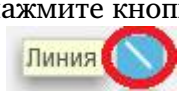


Вот теперь получилось отлично!
Дублируйте Костюм 2.



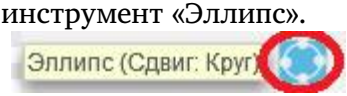

Повторите всё то, что мы делали с Костюмом2, чтобы получилась такая Тётя.

Сейчас готовы все три костюма Тёти, и каждому из них можно вручить самое страшное оружие против котов — веник! Чтобы нарисовать веник, нажмите кнопку «Линия».

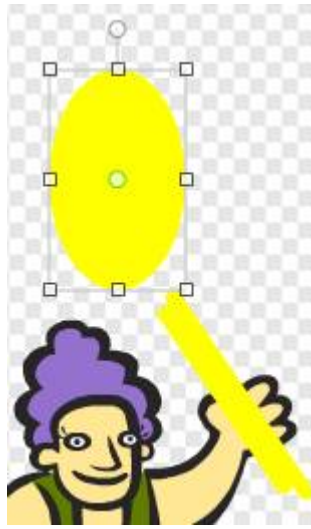



Выберите желтый цвет, немного увеличьте толщину линии и нарисуйте ручку веника.

Затем выберите инструмент «Эллипс».




Нарисуйте жёлтый овал.



Возьмите его за точку вращения, и поверните.

Потом нажмите на овал и передвиньте к ручке веника.

Теперь выберите инструмент «Изменение формы».

Кликните на овал — он станет обведён кружочками — опорными точками. Передвигая опорные точки можно изменять овал.

Кликните на самую верхнюю точку и потяните её с центру овала — форма овала изменится.

Передвиньте следующую точку в центр овала.

Передвиньте третью точку.

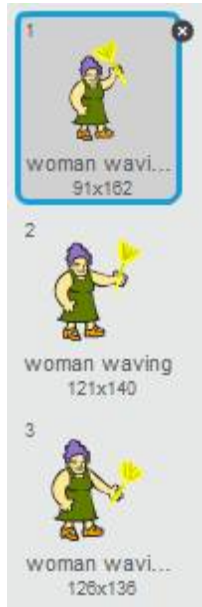
Мы как будто обрезали половину овала! Подрисуйте венику ещё несколько линий для красоты.

Первый костюм готов.

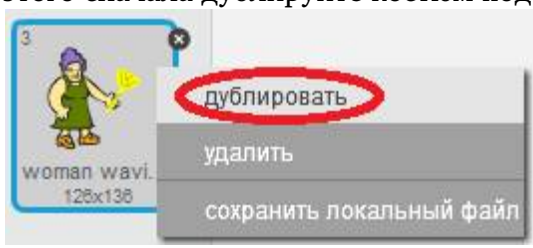
Похожим (аналогичным) способом нарисуйте страшное оружие двум оставшимся костюмам. Костюм2 должен выглядеть так.

А Костюм3 вот так.

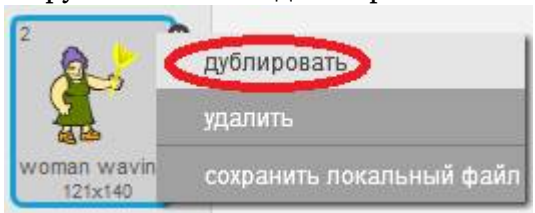
У нас получились три костюма Тёти1. Если их быстро переключать, то будет похоже что она опустила руку с веником.



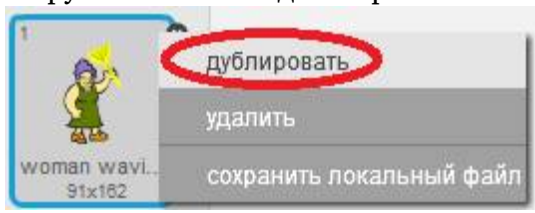

Чтобы анимация была реалистичней, добавим подъём руки с веником. Для этого сначала дублируйте костюм под номером 3.



Потом дублируйте костюм под номером 2.



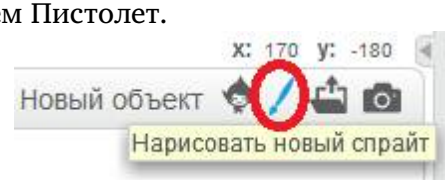

Потом дублируйте костюм под номером 1.




Теперь у нас появились ещё три костюма Тёти1 в которых она поднимает веник. Теперь анимация будет очень реалистичной.

После того, как мы вооружили веником Тётю, настала пора дать секретное оружие Коту. Это оружие ни разу не применялось ни в каких боевых действиях. Кот применит его впервые. Оружие называется Вантуз! В переводе с английского «One Tooz» означает «один туз», это козырное оружие!

Стрелять Вантузом Кот будет из специального Пистолета. Сначала нарисуем Пистолет.



Увеличьте масштаб.

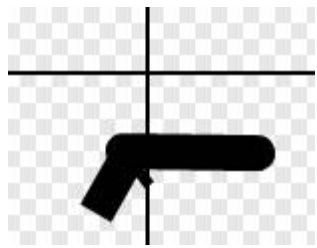
Выберите инструмент Линия.

Увеличьте толщину линии — передвиньте движок правее.

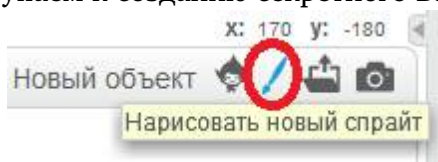
Нарисуйте простейший пистолет.

Теперь очень важное дело — правильно установить точку привязки.

Точка должна быть установлена выше Пистолета, тогда он будет как раз в лапах у Кота. Если установить точку привязки в центр Пистолета, то он окажется у Кота на шее, и стрелять будет неудобно.

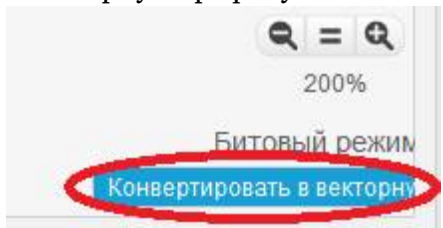


Теперь приступаем к созданию секретного Вантуза.

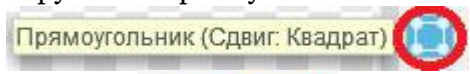



Увеличьте масштаб.

Вантуз будем рисовать в векторном редакторе, так он получится красивее. Справа внизу нажмите кнопку «Конвертировать в векторную графику».

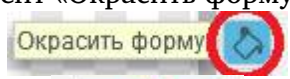



Выберите инструмент «Прямоугольник».



Нарисуйте тонкий прямоугольник — ручку Вантуза.

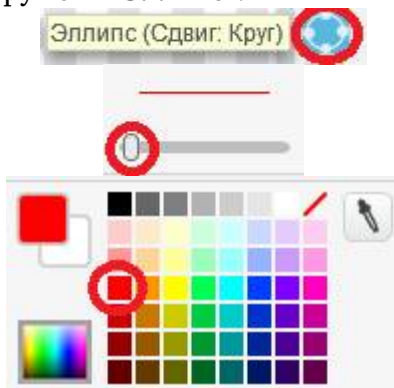
Выберите инструмент «Окрасить форму».



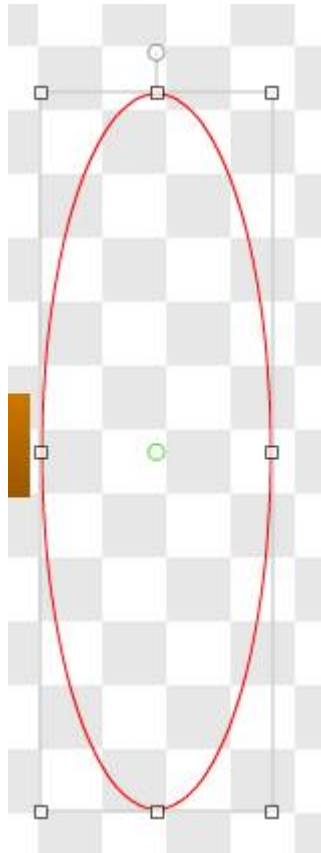
Окрасьте прямоугольник.

Теперь самое интересное. Вантуз имеет сложную форму, поэтому сначала нарисуем просто овал, а потом, с помощью опорных точек, придадим ему необходимую форму. Увеличьте масштаб.

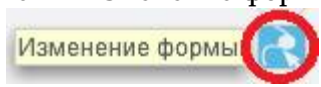
Выберите инструмент «Эллипс».



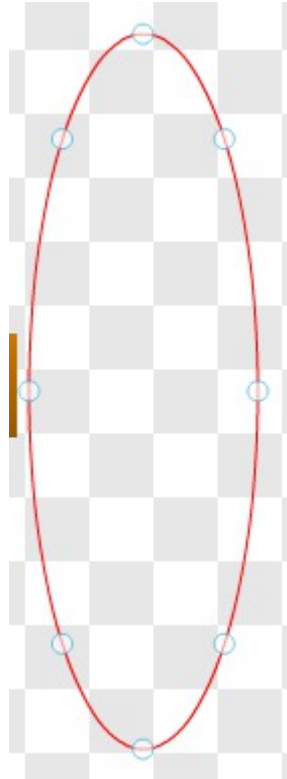

Нарисуйте овал.



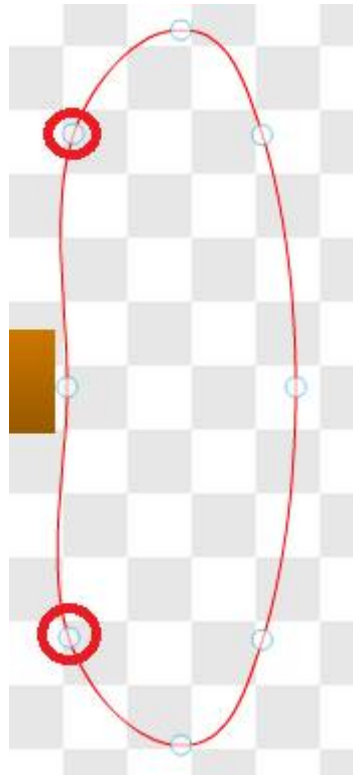
Выберите инструмент «Изменение формы».



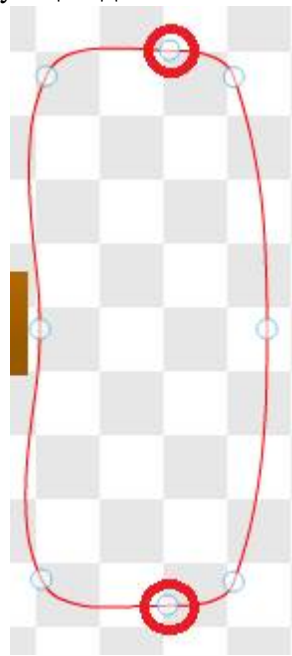

Кликните на овал.



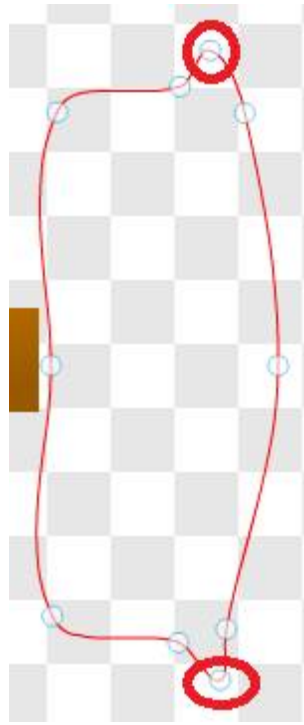
По очереди перемещайте опорные точки, чтобы придать Вантузу его форму. Переместите первые две точки.



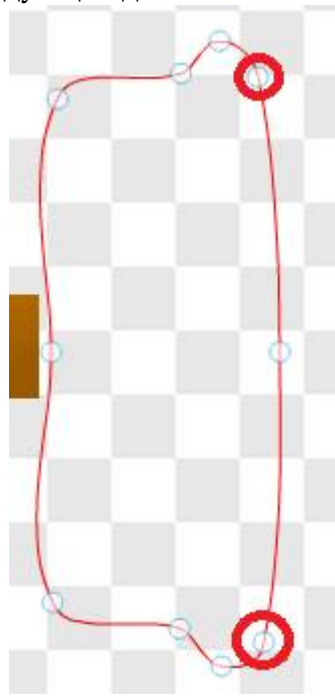
Переместите следующие две точки.



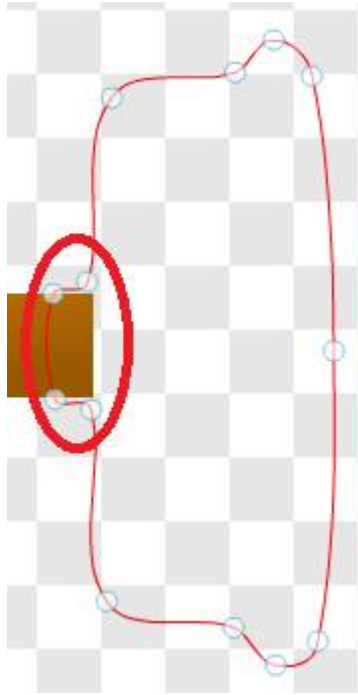
Обратите внимание, если кликнуть в линию между двух точек, то в этом месте появляется новая опорная точка, которую можно двигать. Добавьте две новых опорных точки.



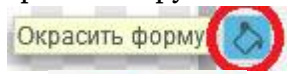
Переместите следующие две точки.

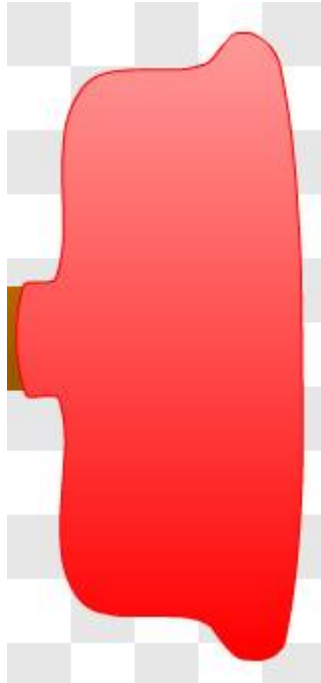


Добавьте ещё три новых опорных точки, и переместите четвёртую как показано на рисунке.

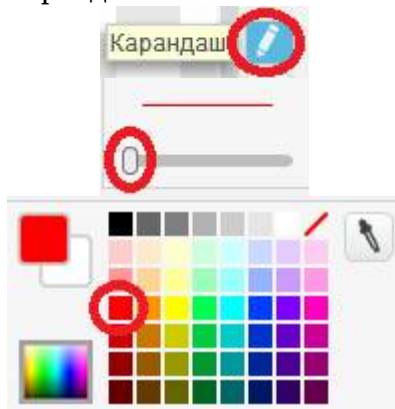


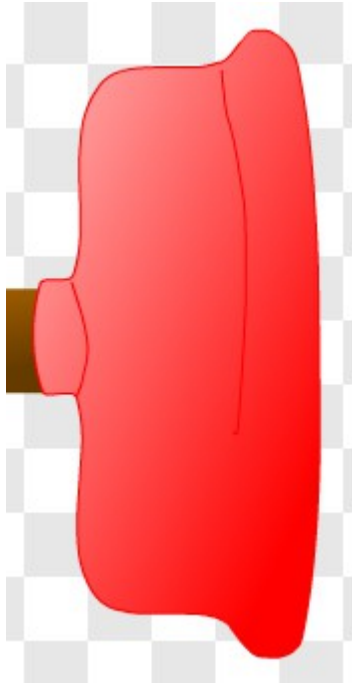
Форма готова! Выберите инструмент «Окрасить форму».



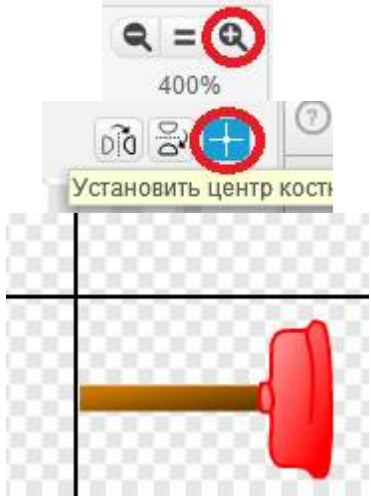


Нарисуем ещё несколько линий для придания Вантузу объёма. Возьмите «Карандаш».

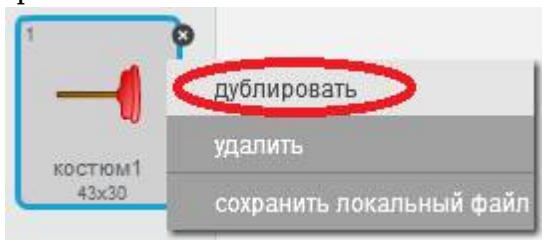




Секретное оружие готово! Осталось только указать правильную точку привязки. Уменьшите масштаб.

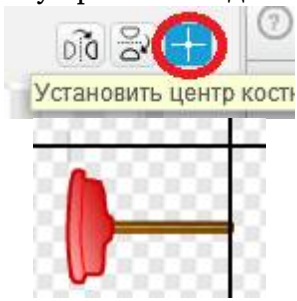



Сделайте второй костюм Вантуза, повёрнутый налево. Дублируйте первый костюм.

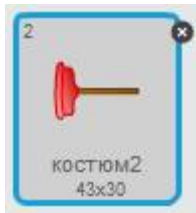


Нажмите кнопку «Отразить слева направо», Вантуз развернётся.

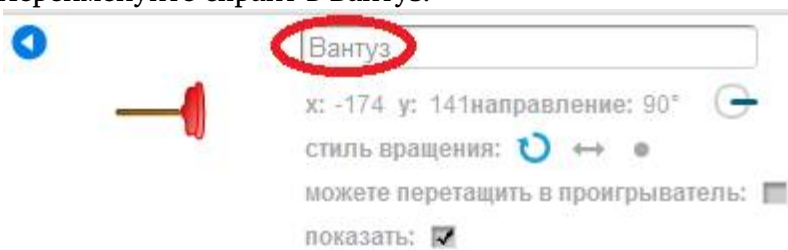
Переустановите точку привязки над концом ручки Вантуза.



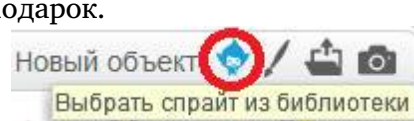

Второй костюм готов.



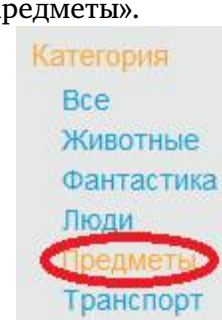

Переименуйте спрайт в Вантуз.




Сейчас добавим самое главное, то, ради чего Кот преодолеет все преграды — Подарок.



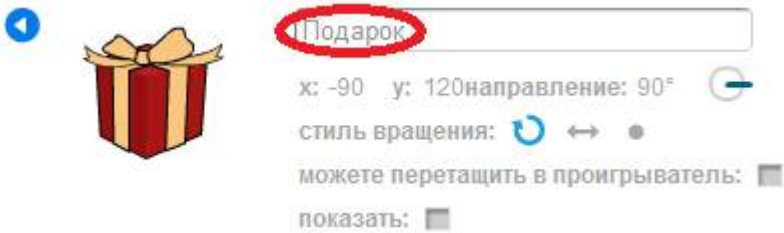

Выберите фильтр «Предметы».




Выберите подарок.

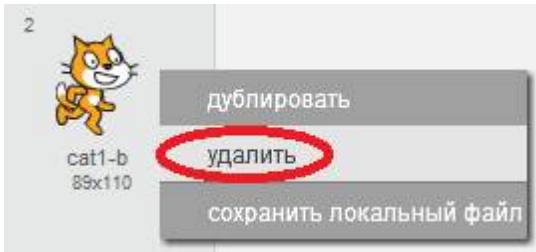
Gift

Переименуйте новый спрайт.

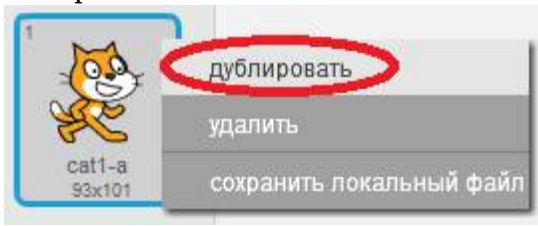



Почти всё готово для начала написания программы. Осталось сделать костюмы для главного героя, для Кота. В процессе игры он не раз получит веником по голове. Надо сделать ему «побитые» костюмы. Изменим ему глазки, носик и рот, Кот будет выглядеть побитым. Сильно за него не переживайте, как только Кот дойдёт до Подарка, всё здоровье восстановится, и он снова улыбнётся жизни. Всего у Кота будет шесть костюмов.

Выберите Кота и откройте графический редактор. Удалите второй костюм.



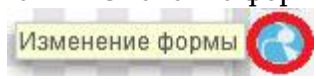

Дублируйте первый костюм.




Выберите второй костюм. Увеличьте масштаб.

Какой красавец!

Кликните инструмент «Изменение формы».

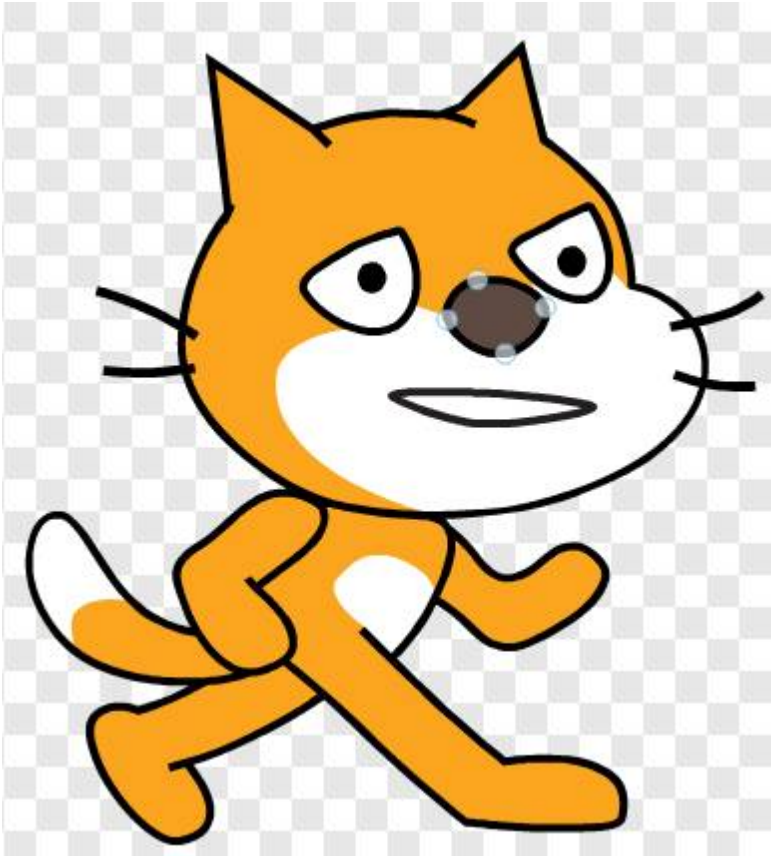



Кликните глаз, и, перемещая опорные точки, уменьшите его.

Второй костюм готов. Дублируйте его. У третьего костюма измените второй глаз.

Третий костюм готов. Дублируйте его. У четвёртого костюма измените рот.

Четвёртый костюм готов. Дублируйте его. У пятого костюма измените нос.

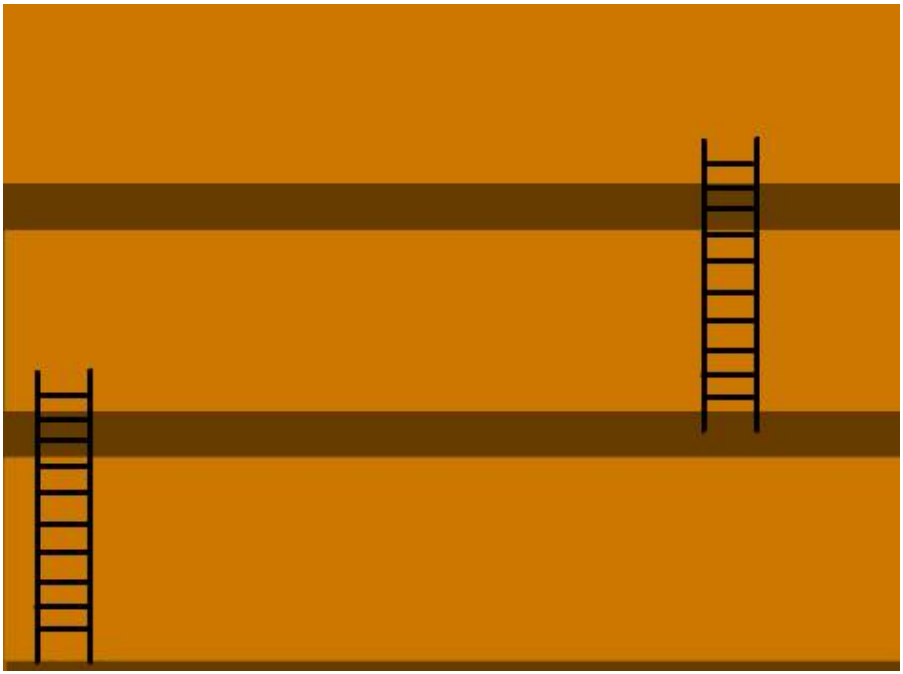


Пятый костюм готов. Дублируйте его. У шестого костюма снова измените рот.

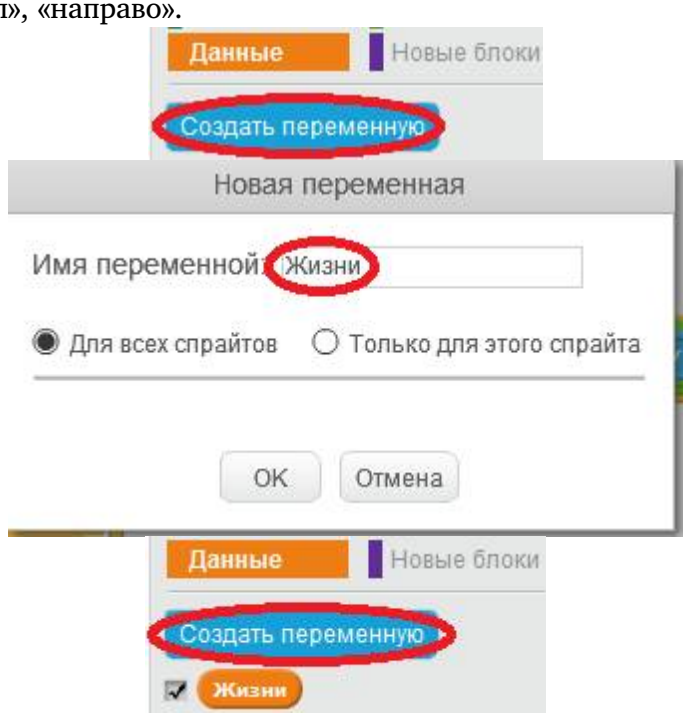
Бедный Котик! Ну и досталось же ему… Я сейчас заплачу…

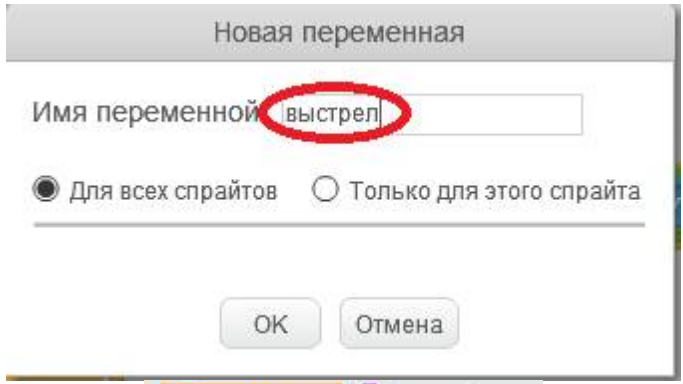
Скорее, выберите первый костюм Кота, пусть он пока улыбается. Встреча с вредными Тётями ещё не скоро.

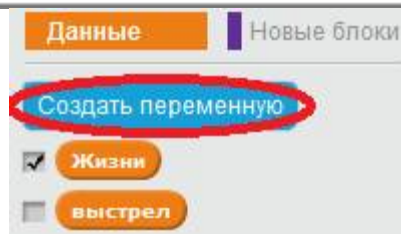
Вот теперь, мы полностью готовы к программированию игры. Для Сцены и для лестниц программа не нужна. Разместите лестницы так, как показано на рисунке.

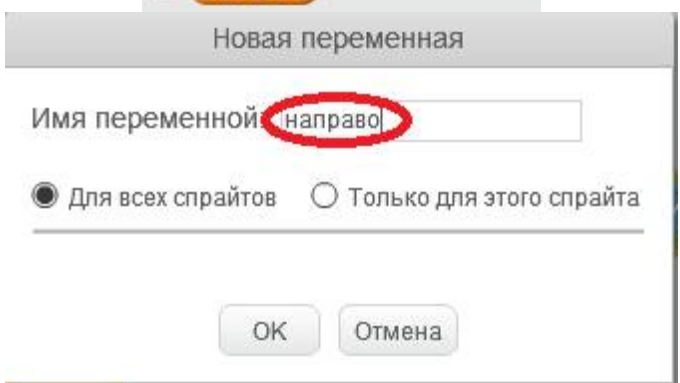


Теперь создадим три глобальных переменных «Жизни», «выстрел», «направо».

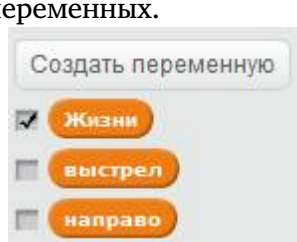




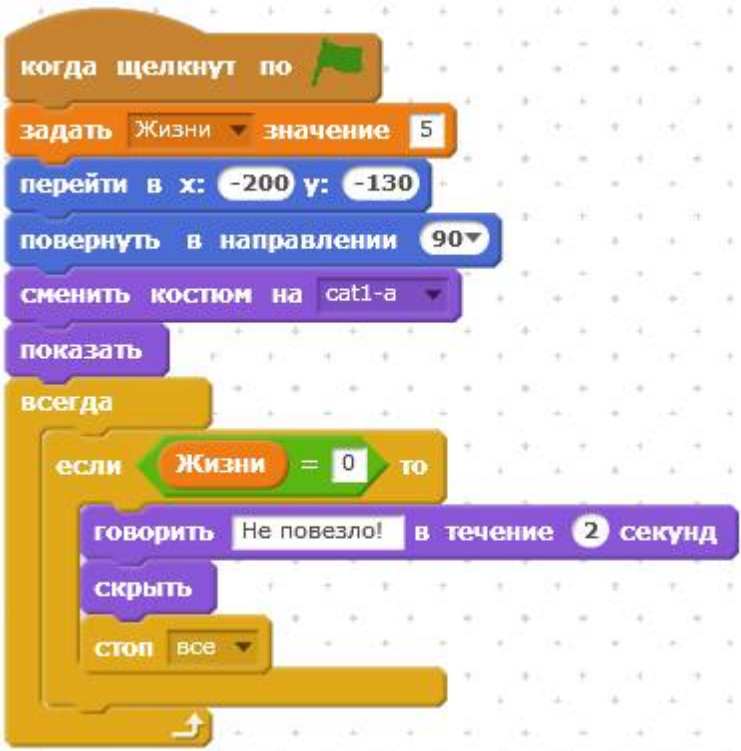

Получились три переменных.




Переменная «Жизни» будет видна на экране, она покажет, сколько жизней осталось у Кота. Переменная «выстрел» будет принимать значение 0 или 1, если Кот выстрелит, то переменная принимает значение 1 и Вантуз полетит. Если «выстрел» = 0, то Вантуз переходит в Пистолет. Переменная «направо» тоже будет принимать только два значения 0 и 1. Она отвечает за направление полёта Вантуза, направо или налево. Такие переменные, как «выстрел» и «направо», которые принимают значение только 0 или 1, называют логическими переменными. У

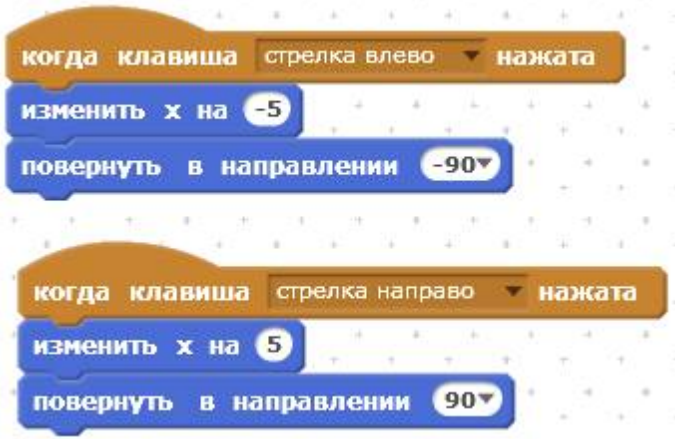
этих переменных очень простая логика: 1 — это «да», 0 — это «нет». Например, если переменная «направо» = 1, то есть «да», значит «да, летим направо». А если «направо» = 0, то есть «нет», «направо не летим, а летим налево».

Начинаем программировать. Сначала сделайте программу Коту. В ней будет шесть скриптов. Первый скрипт. Он задаёт начальные значения, и, потом, всегда следит за значением переменной «Жизни». Если «Жизни» = 0, то конец игры.




Второй скрипт. Он изменяет костюмы Кота. Чем меньше осталось у Кота Жизней, тем страшнее он выглядит.

Ещё два скрипта. Они отвечают за перемещение Котика вправо и влево.


когда клавиша стрелка влево нажата
изменить x на -5
повернуть в направлении -90
когда клавиша стрелка направо нажата
изменить x на 5
повернуть в направлении 90

Следующий скрипт отвечает за лазание по лесенкам вниз. Первое условие, чтобы лазить по первой лесенке, второе — для второй.

когда клавиша стрелка вниз нажата
если касается Лестница1 ? и положение по оси y of Кот > -120 то
повторить 25
изменить y на -5
если касается Лестница2 ? и положение по оси y of Кот > 0 то
повторить 25
изменить y на -5

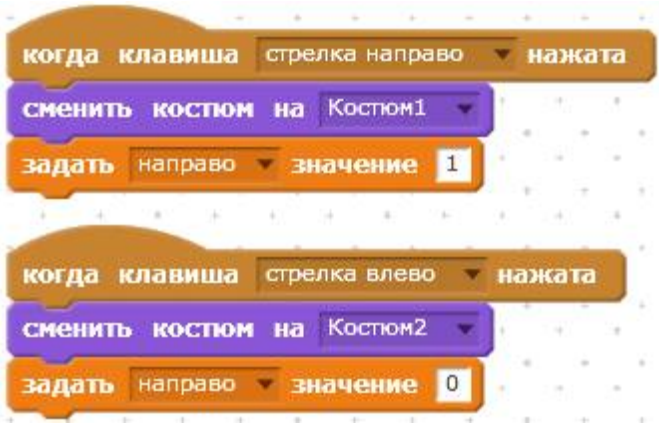
Последний скрипт, чтобы лазить по лесенкам вверх.

когда клавиша стрелка вверх нажата
если касается Лестница1 ? и положение по оси y of Кот < -10 то
повторить 25
изменить y на 5
если касается Лестница2 ? и положение по оси y of Кот < 110 то
повторить 25
изменить y на 5

Теперь сделаем программу для Пистолета. В ней всего три

скрипта. Первый скрипт позволяет Коту взять оружие. Если Кот коснётся Пистолета, то Пистолет всегда будет в лапах у Кота.

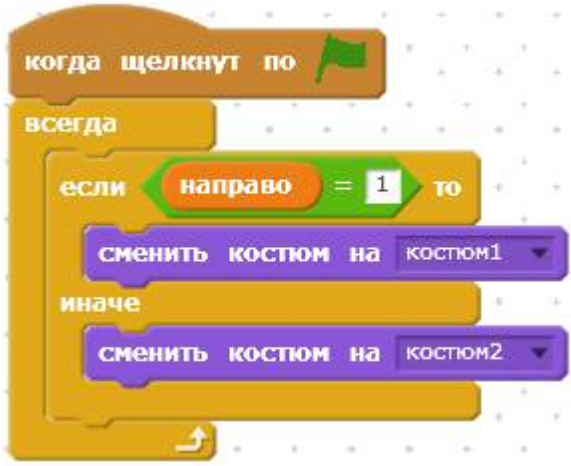
Следующие два скрипта поворачивают Пистолет налево и направо, заодно изменяют значение логической переменной «направо».


когда клавиша стрелка направо нажата
сменить костюм на Костюм1
задать направо значение 1
когда клавиша стрелка влево нажата
сменить костюм на Костюм2
задать направо значение 0

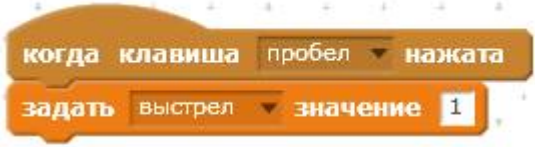
Сейчас запрограммируем Вантуз. Всего у него будет три скрипта. Первый скрипт задаёт начальные значения переменных, и всегда перемещает Вантуз в Пистолет - заряжает Пистолет. А если переменная выстрел равна единице, то Вантуз пролетит 16 раз по 30 точек, а это 480 точек, как раз ширина экрана.

Направление полёта влево или вправо зависит от значения переменной «направо».

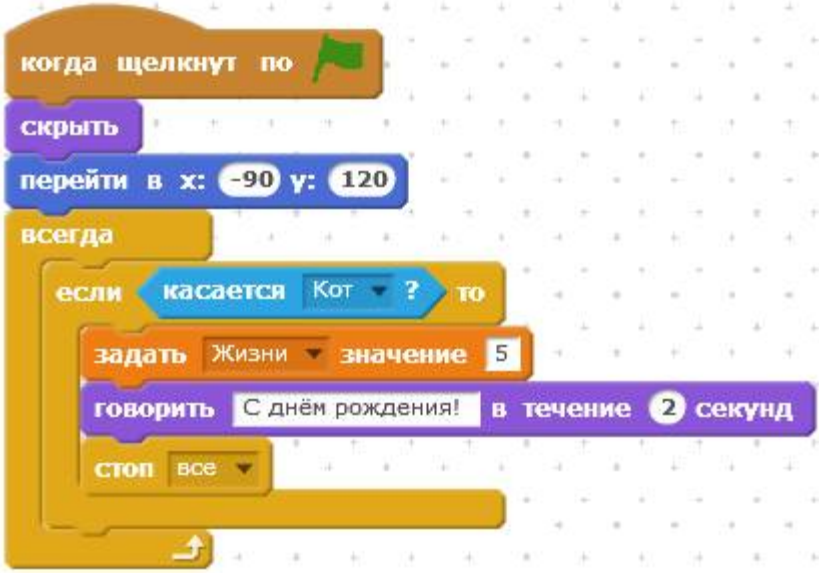
Следующий скрипт отвечает за поворот Вантуза в зависимости от значения переменной «направо».




Третий скрипт позволяет производить выстрел с помощью пробела.




Для Подарка необходимо сделать два скрипта. Первый скрипт будет возвращать Котику Жизни и поздравлять его с днём рождения.

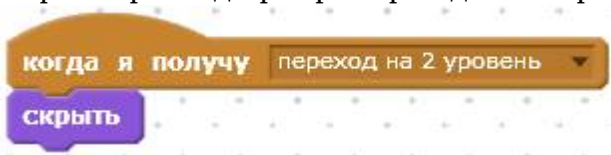



Второй скрипт будет показывать Подарок только на четвёртом уровне.

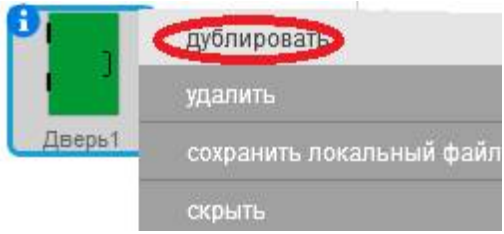
Сейчас у нас только один спрайт с дверью. Это дверь для перехода с первого уровня на второй. Нам нужно ещё две двери, для перехода со второго уровня на третий, и для перехода с третьего уровня на четвёртый. Так как программы всех дверей очень похожи, то сначала сделаем программу для первой двери, и потом два раза её продублируем.

У первой двери будет два скрипта. Первый скрипт передаёт событие «переход на 2 уровень», и меняет костюмы двери при попадании Вантуза.

Второй скрипт прячет дверь при переходе на второй уровень.




Программа Двери1 готова. Нажмите на неё правой кнопкой и выберите «дублировать».




Появится Дверь2 — точная копия Двери1 с такой же программой. Внесите в программу Двери2 небольшие изменения.

Всего у Двери2 будет три скрипта. Первый скрипт такой.

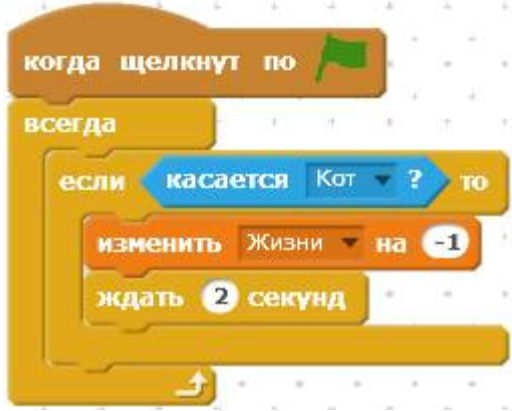
А это второй и третий. Понятно, что на втором уровне дверь появится, а на третьем снова пропадёт.

Теперь дублируйте Дверь2, и сделайте Двери3 три скрипта. Вот первый.

Вот второй и третий.

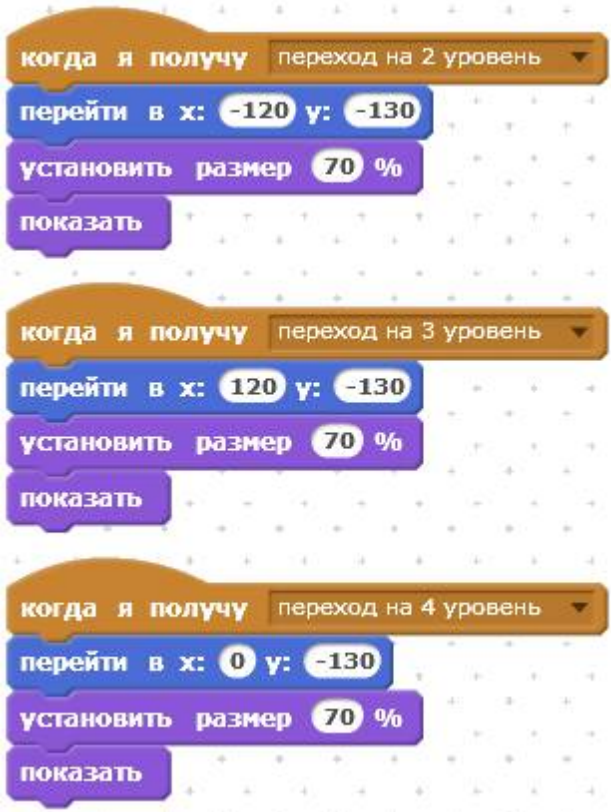
Почти всё готово. Осталось только запрограммировать Тётю1 на ненависть к Коту, и два раза дублировать её. Первый скрипт Тёти1 из-за которого она постоянно бегает влево и вправо по первому этажу.

Второй скрипт Тёти1. В этом скрипте запрограммировано уменьшение Жизней Котика при встрече с вредной Тётей.

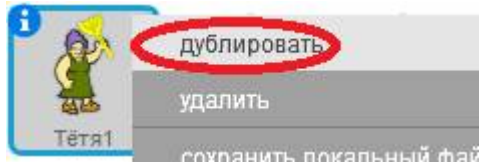



А в следующем скрипте запрограммировано исчезновение Тёти от встречи с секретным оружием. Исчезает она не сразу, сначала сильно уменьшается в размере.

Следующие три скрипта позволяют вредной Тёте оживать на каждом последующем уровне, и с новой силой охотиться за Котом.


когда я получу переход на 2 уровень
перейти в x: -120 y: -130
установить размер 70 %
показать
когда я получу переход на 3 уровень
перейти в x: 120 y: -130
установить размер 70 %
показать
когда я получу переход на 4 уровень
перейти в x: 0 y: -130
установить размер 70 %
показать

Всего получилось шесть скриптов Тёти1. Нажмите на неё правой кнопкой мышки и продублируйте её.

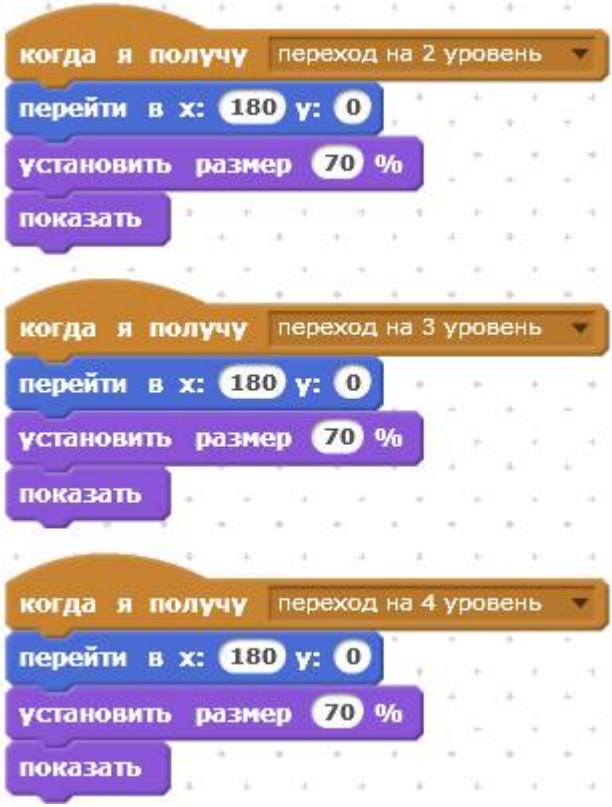



Появится спрайт Тётя2 с такой же программой как у Тёти1. Измените скрипты Тёти2. Первый скрипт должен стать таким. Изменить надо только координаты в блоке «перейти» и количество повторений в цикле.

Второй и третий скрипты ничем не отличаются. Вот второй.

Вот третий.

Следующие три скрипта отличаются.

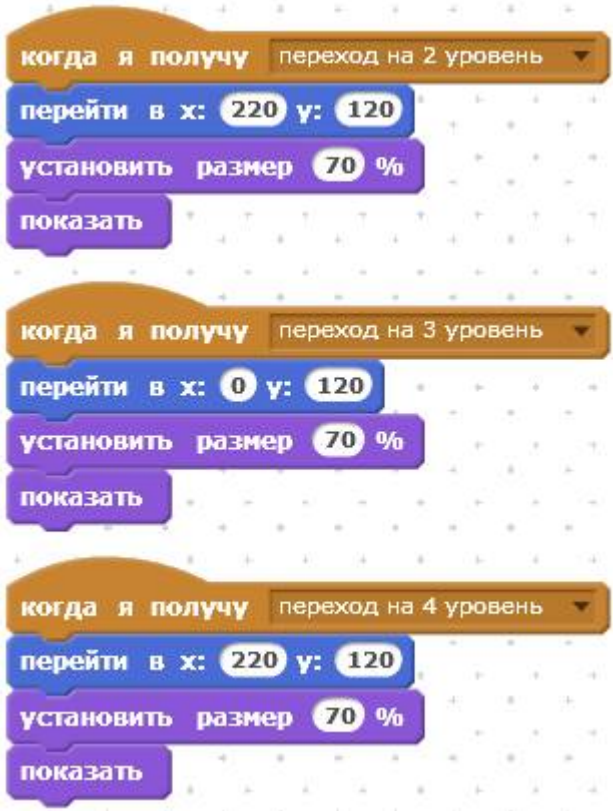

когда я получу переход на 2 уровень
перейти в x: 180 y: 0
установить размер 70 %
показать
когда я получу переход на 3 уровень
перейти в x: 180 y: 0
установить размер 70 %
показать
когда я получу переход на 4 уровень
перейти в x: 180 y: 0
установить размер 70 %
показать

Снова дублируйте Тётю1. Появится Тётя3, вот её первый скрипт. Изменить надо только координаты в блоке «перейти».

Второй и третий скрипты опять без изменений. Вот второй.

Вот третий.

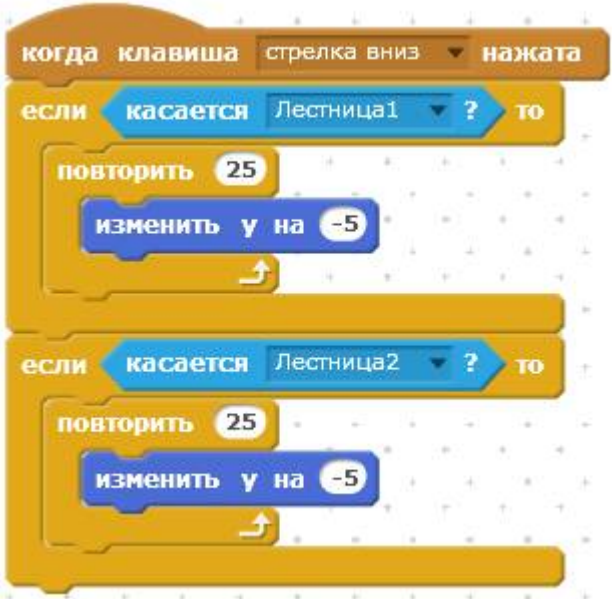
Вот и последние три скрипта Тёти3.


когда я получу переход на 2 уровень
перейти в x: 220 y: 120
установить размер 70 %
показать
когда я получу переход на 3 уровень
перейти в x: 0 y: 120
установить размер 70 %
показать
когда я получу переход на 4 уровень
перейти в x: 220 y: 120
установить размер 70 %
показать

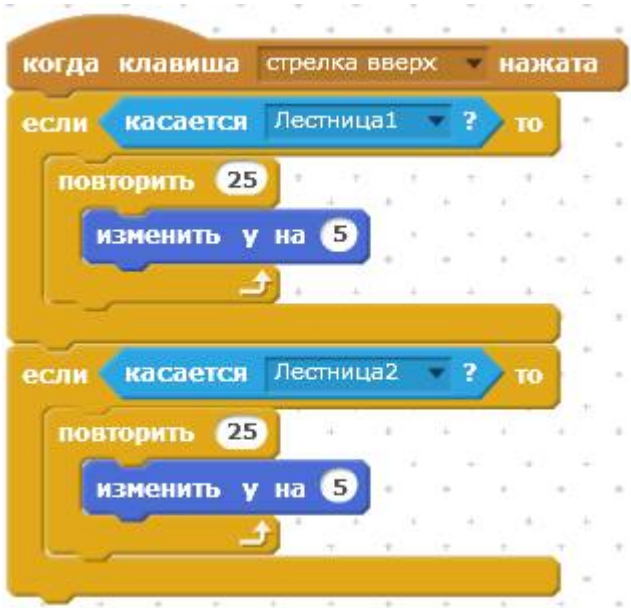
Программа готова.

Вопросы по игре «Битва за день рождения Кота».

1. Как увеличить скорость полёта Вантуза?
2. Как увеличить количество жизней у Кота?
3. Какой баг появится, если мы упростим скрипты лазания Кота по лесенкам, удалив условие высоты? Сначала попробуйте догадаться без экспериментов, а потом, для самопроверки, измените программу Кота и посмотрите, что будет. Для проверки Вашего ответа измените скрипт для стрелки вниз как показано на этом рисунке.




А вот так измените скрипт для стрелки вверх.




Задания по доработке игры «Битва за день рождения Кота».

1. Сделайте так, чтобы при выстреле из пистолета вылетало немного порохового дыма.

2. Добавьте к игре ещё один уровень.

3. Добавьте несколько спрайтов с лекарством для Кота. Если Кот коснётся лекарственного спрайта, то его здоровье увеличится на 1.

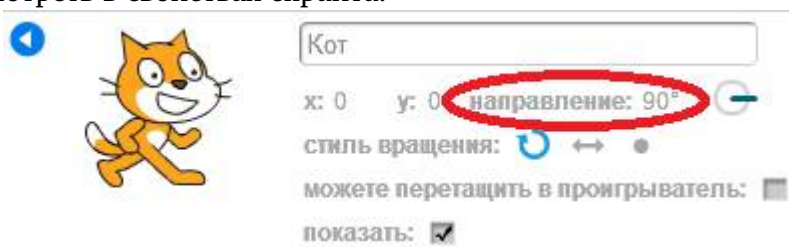
# ПРИЛОЖЕНИЕ 1. ОСНОВНЫЕ ПОНЯТИЯ SCRATCH 2.

В этом приложении кратко объяснены основные понятия Scratch 2:

- Направление;
- Выпадающий список;
- Стили вращения;
- Пронумерованные цвета;
- Графические эффекты;
- Слои.

А. НАПРАВЛЕНИЕ.

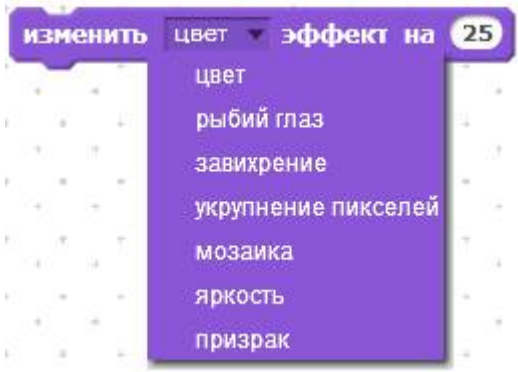
Под направлением спрайта понимается направление движения спрайта, в котором он будет перемещаться при использовании блока «Идти». Текущее направление можно посмотреть в свойствах спрайта.




Направление измеряется в градусах. Направлению в ноль градусов соответствует движение вертикально вверх. Направление 90 градусов — движение вправо, направление -90 градусов — движение влево. Направление 180 градусов — движение вниз. Подробнее о градусах можно узнать в приложении 3.

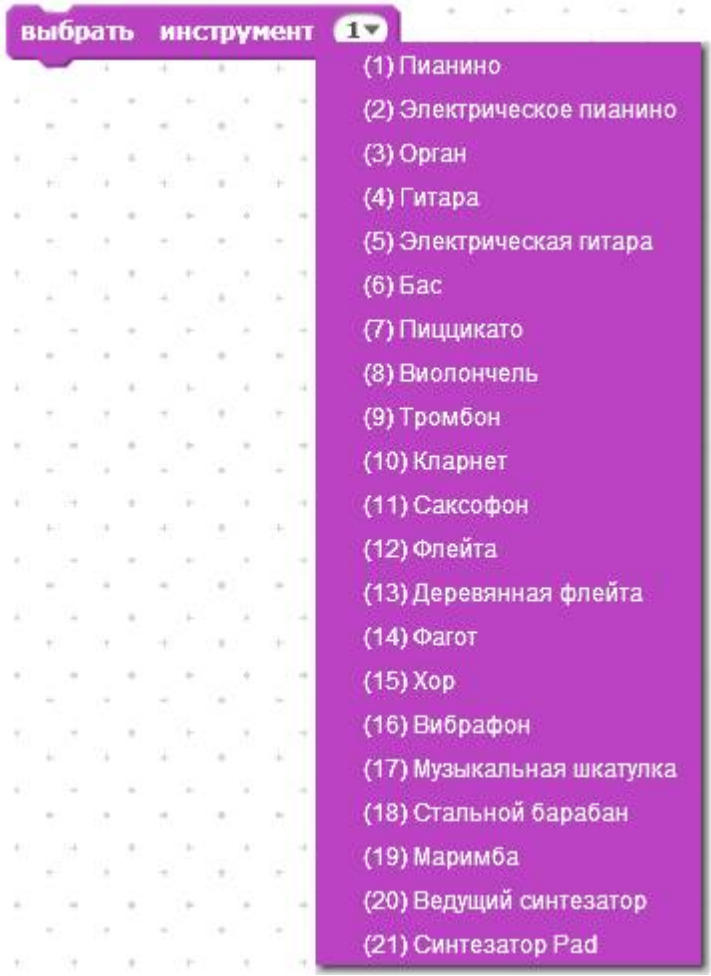
Б. ВЫПАДАЮЩИЙ СПИСОК.

Некоторые блоки позволяют выбирать варианты с помощью выпадающего списка. Например, блок изменения эффектов позволяет выбрать один из семи эффектов.




Блок управления спрайтом с клавиатуры позволяет выбрать, на какую клавишу реагировать. Выпадающий список очень длинный, включает в себя все буквы английского алфавита, цифры, стрелки и пробел.

Блок выбора музыкального инструмента позволяет выбрать один из двадцати одного музыкального инструмента

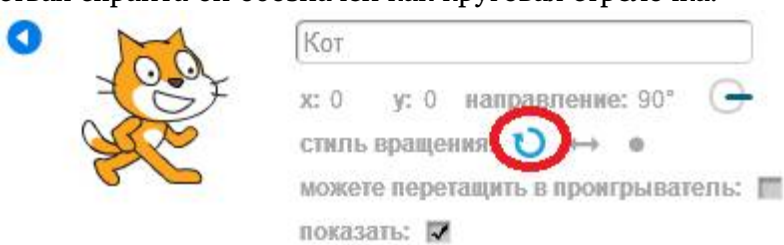



В. СТИЛИ ВРАЩЕНИЯ.

В Scratch 2.0 существует всего три типа вращения. Рассмотрим каждый из них.

*Круговой стиль вращения.*

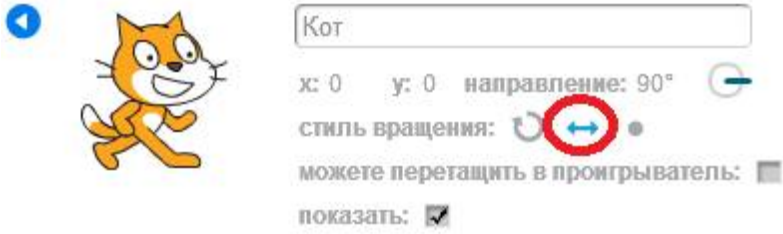
Спрайт может поворачиваться в любом направлении. В свойствах спрайта он обозначен как круговая стрелочка.




При изменении текущего направления спрайта, спрайт будет поворачиваться на требуемое количество градусов.

*Стиль вращения вправо-влево.*

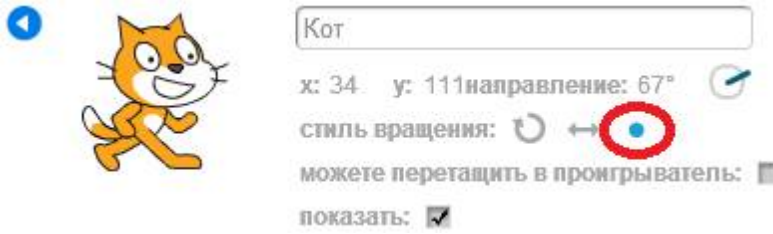
Спрайт может поворачиваться только в двух направлениях. Направо или налево. Поворот в остальных направлениях невозможен. В свойствах спрайта он обозначается как двойная стрелочка.




При изменении текущего направления спрайта, спрайт не будет поворачиваться на требуемое количество градусов, как в круговом стиле вращения. При текущем направлении спрайта от 0 до 180 градусов спрайт будет повернут вправо, при направлении от 0 до -180 градусов, спрайт будет повёрнут влево.

*Не вращать.*

При выборе этого стиля вращения спрайт не будет вращаться. В свойствах спрайта он обозначен как кружочек.

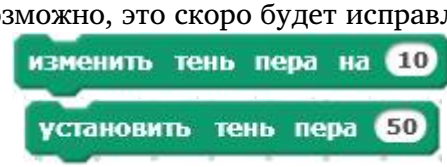



При изменении текущего направления спрайта вращаться он не будет.

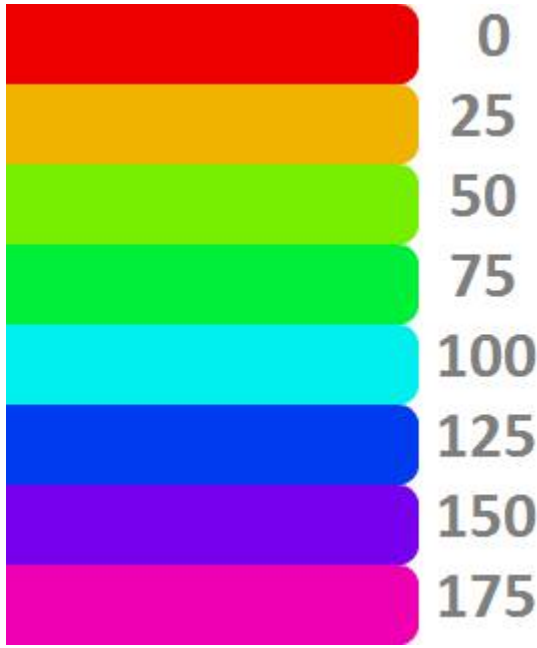
Г. ПРОНУМЕРОВАННЫЕ ЦВЕТА И ОТТЕНКИ.

В Scratch каждый цвет имеет свой номер. Оттенки цвета также имеют номера. Это сделано для более удобной работы с цветом. Числовые значения цветов используются в блоках «изменить цвет пера» и «установить цвет пера».

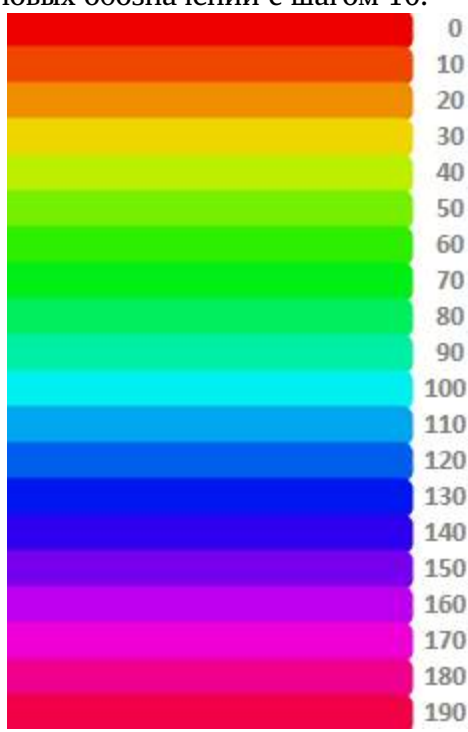
Числовые значения оттенков используются в блоках «изменить тень пера» и «установить тень пера». Обратите внимание, из-за ошибки перевода на русский язык оттенок пера назван тенью, возможно, это скоро будет исправлено.



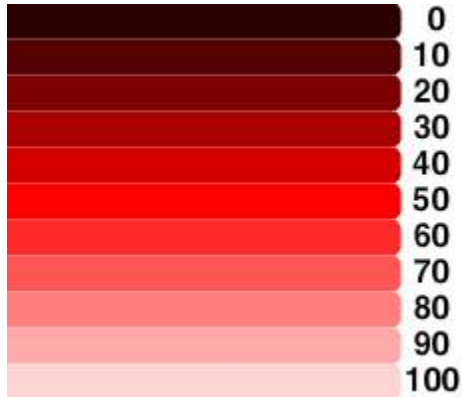
Началом отсчёта, цветом с номером ноль, как и в радуге, является красный цвет. На следующем рисунке приведена таблица соответствия цветов и их числовых обозначений с шагом 25.




Ниже приведена более подробная таблица соответствия цветов и их числовых обозначений с шагом 10.



Все оттенки цветов также пронумерованы от ноля, самого тёмного оттенка, до ста, самого светлого оттенка. Ниже приведена таблица оттенков. Нормальный красный цвет получается при значении оттенка в 50.

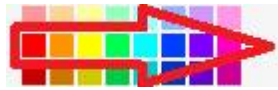



Д. ГРАФИЧЕСКИЕ ЭФФЕКТЫ.

В Scratch 2.0 есть семь графических эффектов, которые могут быть применены к спрайтам и фонам. Графические эффекты применяются с использованием блоков «Изменить эффект на», и «Установить эффект в значение». Вид графического эффекта выбирается с помощью выпадающего списка, значение эффекта вводится с клавиатуры. Блок «Убрать графические эффекты», как следует из его названия, убирает все графические эффекты.

*Эффект «цвет».*

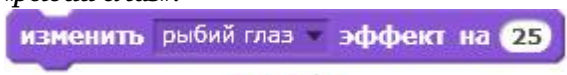
Это циклический эффект. Цвет спрайта изменяется при изменении значения эффекта от 0 до 200. При значении больше 200, например, при 201, спрайт выглядит так же, как и при 1. При 300, так же, как и при 100. При 400, так же как при 200 и при 0, то есть, без изменения цвета. Например, если у нас есть спрайт красного цвета, то при значении эффекта «цвет» равного 0, спрайт будет красным. При значении эффекта 25, спрайт станет оранжевым. При значении эффекта 50, спрайт станет жёлтым и так далее, пока, при достижении значения эффекта 200, спрайт снова не станет красным. Направление изменения цвета спрайта показано на рисунке.



Если спрайт изначально был не красного, а, например, как Кот, оранжевого цвета, то при значении эффекта 25 он станет жёлтым, при 50 зелёным, а при 75 — голубым. Изменение цветов будет происходить в том же порядке, и, при достижении значения эффекта 200, спрайт снова станет оранжевым.

Числовые обозначения цветов также используются при рисовании пером. Ниже приведена таблица соответствия цветов и их числовых обозначений с шагом 25.

*Эффект «рыбий глаз».*



При применении этого эффекта спрайт как будто надувается изнутри. Края спрайта остаются почти без изменений, а центр увеличивается в размере.

Эффект не циклический, значение эффекта может быть очень велико, однако, при значении эффекта больше 1000, спрайт практически не изменяется, он превращается в круглое цветное пятно. Самые интересные изменения происходят со спрайтом при изменении значения эффекта «рыбий глаз» от 0 до 200.

На рисунке показан Кот со значением эффекта «рыбий глаз» равным 200.

*Эффект «завихрение».*

При применении этого эффекта спрайт как будто закручивается изнутри. Края спрайта остаются почти без изменений, а центр вращается против часовой стрелки. Эффект не циклический, значение эффекта может быть очень велико, однако, при значении эффекта больше 3000, спрайт практически не изменяется, он превращается в круглое разноцветное пятно. Самые интересные изменения происходят со спрайтом при изменении значения эффекта «завихрение» от 0 до 500. На рисунке показан Кот со значением эффекта «рыбий глаз» равным 500.

*Эффект «укрупнение пикселей».*

Пиксели — это точки на экране из которых состоит изображение спрайта.

При применении эффекта «укрупнение пикселей», точки изображения начинают увеличиваться в размерах и сливаться, образуя всё более крупные квадратики. В результате, при значении эффекта равном 500, остаётся один или два цветных квадратика.

*Эффект «мозаика».*

Этот эффект превращает спрайт в мозаику. Изменения происходят не плавно, а скачкообразно. Например, при значении эффекта от 0 до 4, со спрайтом ничего не происходит. При значении 5 спрайт превращается в мозаику из четырёх элементов. При дальнейшем изменении значения от 5 до 14 спрайт по-прежнему состоит из четырёх элементов, а при достижении 15 количество котиков скачкообразно увеличивается до девяти.

При достижении значения эффекта «мозаика» 25 котиков станет 16.

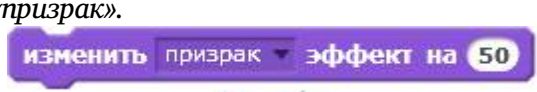
При достижении значения эффекта «мозаика» 35 котиков станет 25.

При дальнейшем увеличении значения эффекта котиков уже невозможно разобрать, спрайт превращается в набор цветных точек. Вот так будет выглядеть Кот при применении к нему эффекта «мозаика» со значением 200.

*Эффект «яркость».*

При применении этого эффекта спрайт становится ярче. При значении эффекта больше 100 никакие изменения не происходят, потому что спрайт достиг максимальной яркости. На рисунке показан Кот с яркостью 100. Все линии чёрного цвета стали белыми.

*Эффект «призрак».*



Этот эффект можно было бы назвать «прозрачность», он задаёт величину прозрачности спрайта. При значении эффекта «призрак» равного 0, прозрачности нет совсем, а при значении 100, прозрачность максимальна, то есть спрайт вообще невидим.

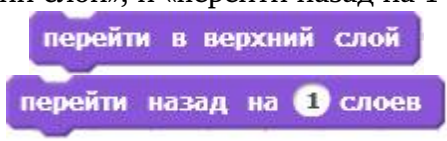
Е. СЛОИ.

Слои определяют взаимное расположение спрайтов на сцене. Если один из спрайтов расположен в верхнем слое, а другой на один слой ниже, то спрайт, расположенный в верхнем слое будет виден целиком, а спрайт, расположенный на слой ниже, будет из-за него выглядывать. Сцена всегда находится в самом нижнем

слое. В данном примере товарищ Гобо находится в верхнем слое, а гражданин Кот на слой ниже.

Так получилось из-за того, что Гобо был добавлен позднее. Каждый новый спрайт, добавленный в проект, располагается в самом верхнем слое. Добавленная последней синяя Собака расположена в третьем слое, выше, чем товарищ Гобо.

Для управления расположением спрайтов служат блоки «перейти в верхний слой», и «перейти назад на 1 слоев».



# ПРИЛОЖЕНИЕ 2. СПРАВОЧНИК ПО БЛОКАМ SCRATCH 2.0.

В этом приложении перечислены все блоки Scratch 2 с кратким описанием.

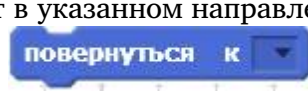
А. СИНИЕ БЛОКИ ДВИЖЕНИЯ.

Спрайт переместится на 10 шагов в текущем направлении.

Повернуть спрайт на 15 градусов по часовой стрелке по отношению к текущему направлению спрайта.

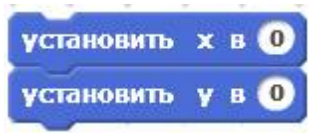
Повернуть спрайт на 15 градусов против часовой стрелки по отношению к текущему направлению спрайта.

Повернуть спрайт в указанном направлении.

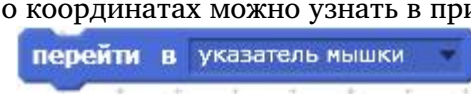



Повернуться по направлению к одному из спрайтов. Для выбора спрайта следует нажать на чёрный треугольничек, и выбрать из выпадающего списка.

Переместить спрайт в точку экрана с заданными координатами X и Y. В точке с заданными координатами окажется центр спрайта. Применение этого блока равносильно применению одновременно двух блоков «Установить X в» и «Установить Y в».




Подробнее о координатах можно узнать в приложении 4.




Переместить спрайт в указатель мышки, либо в центр указанного, с помощью выпадающего списка, спрайта. При этом центр спрайта окажется на самом кончике мышки.

Плавно переместить спрайт в точку с заданными координатами. Время перемещения задаётся в секундах. Для быстрого перемещения можно использовать десятичные дроби. Например, 0.5 секунд — это половина секунды, 0.25 секунд — это четверть секунды, 0.1 секунд — это одна десятая секунды, это очень быстро.

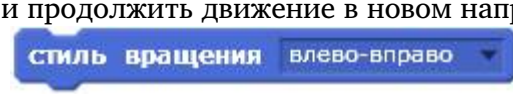
Изменить координату X спрайта на указанную величину. При этом спрайт сместиться вправо. При изменении на отрицательную величину, спрайт сместиться влево.

Установить новое значение координаты X спрайта.

Изменить координату Y спрайта на указанную величину. При этом спрайт сместиться вверх. При изменении на отрицательную величину, спрайт сместиться вниз.

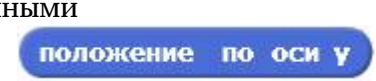
Установить новое значение координаты X спрайта.

Очень важный блок! Позволяет спрайту отразиться от границы окна и продолжить движение в новом направлении.




Изменяет стиль вращения спрайта. Всего есть три типа вращения: вправо-влево, не вращать, кругом. Подробнее о стилях вращения можно узнать в приложении 1. (вставить ссылку)

Это не управляющий блок, а блок получения информации из программы. Он возвращает из памяти работающей программы текущее значение координаты X спрайта. Этот блок, как и все овальные блоки, можно использовать в операторах наравне с числами и переменными



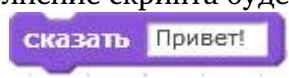

Блок получения информации из программы. Он возвращает из памяти работающей программы текущее значение координаты Y спрайта.

Блок получения информации из программы. Он возвращает из памяти работающей программы текущее направление спрайта в градусах.

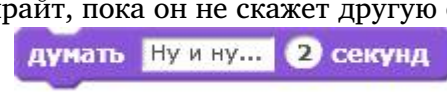
Совет. При двойном клике на любом из блоков, текущий спрайт выполнит указанную команду.

Б. ФИОЛЕТОВЫЕ БЛОКИ ВНЕШНОСТИ.

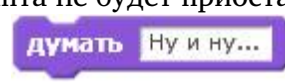
Спрайт будет говорить слово «Привет!» в течение двух секунд. При этом выполнение скрипта будет приостановлено.



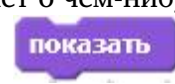

Спрайт постоянно будет говорить слово «Привет!» без остановки выполнения скрипта. Сказанная фраза будет постоянно сопровождать спрайт, пока он не скажет другую фразу.




Спрайт будет думать «Ну и ну…» в течение двух секунд. При этом выполнение скрипта не будет приостановлено.



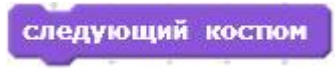

Спрайт постоянно будет думать «Ну и ну…» без остановки выполнения скрипта. Эта мысль будет постоянно сопровождать спрайт, пока он не подумает о чем-нибудь другом.




Спрайт станет виден. Если, при выполнении этого блока, спрайт не виден, то, возможно, он скрыт другим спрайтом. В этом случае может помочь блок «Перейти в верхний слой». Также следует проверить координаты спрайта, может быть, он находится за границей экрана.

Спрайт станет невидимым.

Спрайт изменит свой текущий костюм на костюм, выбранный из выпадающего списка.




Спрайт изменит свой текущий костюм на следующий костюм.

Текущий фон сцены изменится на фон, выбранный из выпадающего списка.

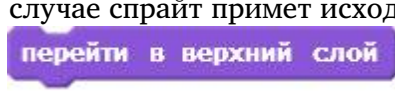
Значение эффекта «цвет» изменится на указанную величину. В этом примере значение эффекта увеличится на 25.

Значение эффекта «цвет» изменится до указанного значения. В этом примере значение эффекта станет равно нулю.

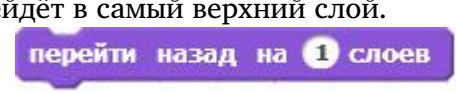
Спрайт примет исходный вид, все графические эффекты будут убраны.

Изменить размер спрайта на указанное количество процентов. В данном случае на 10%.

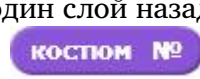
Установить размер спрайта в процентах от исходного размера. В данном случае спрайт примет исходный размер.




Спрайт перейдёт в самый верхний слой.

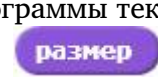



Спрайт перейдёт на один слой назад.




Это не управляющий блок, а блок получения информации из программы. Он возвращает из памяти работающей программы цифровое значение текущего номера костюма.

Блок получения информации из программы. Он возвращает из памяти работающей программы текущее имя фона.




Блок получения информации из программы. Он возвращает из памяти работающей программы текущий размер спрайта.

У сцены есть ещё несколько дополнительных блоков, которых нет у спрайтов.

Сцена сменит фон на выбранный из выпадающего списка.

Фон сцены изменится на следующий.

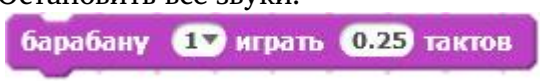
Блок получения информации из программы. Он возвращает из памяти работающей программы цифровой номер текущего фона.

В. МАЛИНОВЫЕ БЛОКИ ЗВУКОВ.

Играть звук «pop», выбранный из библиотеки звуков, без остановки выполнения скрипта.

Играть звук «pop», выбранный из библиотеки звуков до завершения. При этом выполнение скрипта будет приостановлено.

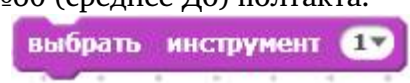
Тишина! Остановить все звуки.




Первому барабану играть одну четверть такта. В выпадающем списке можно выбрать один из восемнадцати барабанов.

Подождать одну четверть такта.

Играть ноту №60 (среднее До) полтакта.




Выбрать один из двадцати музыкальных инструментов с помощью выпадающего списка.

Изменить громкость на указанное количество процентов. В данном случае сделать тише на 10%.

Установить требуемый уровень громкости в процентах от системного уровня.

Блок получения информации из программы. Он возвращает из памяти работающей программы текущий уровень громкости в процентах.

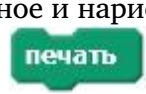
Изменить темп на 20 ударов в минуту.

Установить темп 60 ударов в минуту.

Блок получения информации из программы. Он возвращает из памяти работающей программы текущее значение темпа в ударах в минуту.

Г. ЗЕЛЁНЫЕ БЛОКИ РИСОВАНИЯ ПЕРОМ.

Очищает всё отпечатанное и нарисованное перьями на сцене.




Отпечатывает изображение спрайта на сцене. Не на фоне, а на сцене. Фоны можно изменять, а отпечаток будет поверх всех фонов.

Опустить перо. После этой команды все перемещения пера будут сопровождаться линией.

Поднять перо. После этой команды перемещения пера не будут оставлять следов на сцене.

Установить цвет пера. Выбор цвета можно осуществить при помощи курсора.

Изменить цвет пера на 10. Принцип изменения цвета такой же, как и в блоке изменения графического эффекта «цвет».

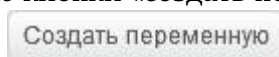
Установить цвет пера в соответствии с таблицей цветов.

Изменить оттенок пера на 10%.

Установить оттенок пера в 50%.

Д. ОРАНЖЕВЫЕ БЛОКИ ДАННЫХ.

Блоки переменных не видны до тех пор, пока вы не создадите переменную с помощью кнопки «создать переменную».



Это сама переменная. При использовании в программе этот миниблок возвращает из памяти работающей программы текущее значение переменной «а». Если около переменной поставить маленькую галочку,

то её значение будет показано на экране.

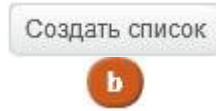
Установить значение переменной «а» в определённое значение. Значение может быть, как числовым, так и текстовым. С помощью выпадающего списка можно выбрать любую переменную.

Изменить значение переменной «а» на единицу. С помощью выпадающего списка можно выбрать любую переменную. Изменять можно только числовую переменную, при попытке изменить текстовую переменную будет ошибка «NaN».

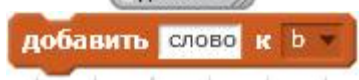
Переменная будет показана на экране, и около неё появится галочка.

Переменная скроется из вида, а маленькая галочка будет снята.

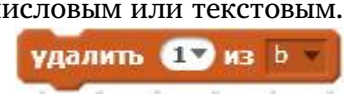
Блоки списков не видны до тех пор, пока вы не создадите список с помощью кнопки «создать список».



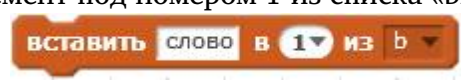

Это сам список. Если около него поставить маленькую галочку, то на экране появится список.




Добавить ещё один элемент списка к списку «b». Элемент списка может быть числовым или текстовым.




Удалить элемент под номером 1 из списка «b».




Вставить значение в элемент списка «b» под номером 1, при этом все последующие элементы сдвинуться. Вставляемое значение может быть либо числовым, либо текстовым.

Заменить значение в элементе списка «b» под номером 1, при этом все последующие элементы останутся на своих местах. Вставляемое значение может быть либо числовым, либо текстовым.

Блок возвращает из памяти работающей программы текущее значение элемента списка «b» под номером 1.

Блок возвращает из памяти работающей программы длину списка «b», то есть количество элементов списка «b».

Логическое значение. Принимает значение «истина», если список «b» содержит элемент со значением «слово».

Список будет показан на экране, и около него появится галочка.

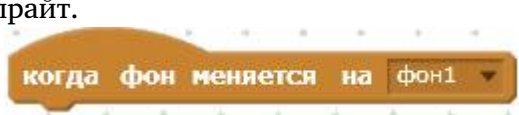
Список скроется из вида, а маленькая галочка будет снята.

Е. БЛОКИ СОБЫТИЙ.

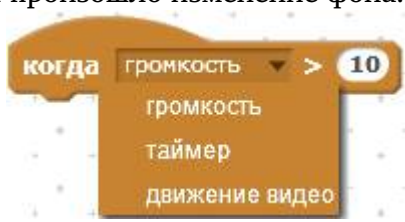
Блок начала выполнения программы. Выполняется, если нажат зелёный флажок.

Блок управления с клавиатуры. Выполняется, если нажата клавиша, выбранная из выпадающего списка.

Блок управления мышкой. Выполняется, если мышкой кликнуть на спрайт.




Блок управления скриптом в зависимости от фона. Выполняется, если произошло изменение фона.

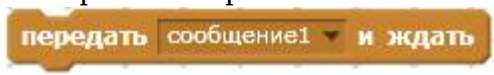



Блок управления скриптом в зависимости от уровня громкости, либо от значения таймера, либо от движения видео. Выполняется при соответствующем изменении параметра, выбранного из выпадающего списка.

Блок управления скриптом с помощью сообщений. Выполняется при получении соответствующего сообщения,

выбранного из выпадающего списка.

Блок передачи сообщения, на которое реагирует блок «когда я получу сообщение». При передаче сообщения с помощью этого блока выполнение скрипта не приостанавливается.




Блок передачи сообщения, на которое реагирует блок «когда я получу сообщение». При передаче сообщения с помощью этого блока выполнение скрипта приостанавливается, и он ждёт, пока другие спрайты выполнят все скрипты блока «когда я получу сообщение» до конца. Затем выполнение текущего скрипта продолжается. Чтобы лучше понять, как работает этот блок, потренируемся на кошках. Сделайте Коту следующие два скрипта.

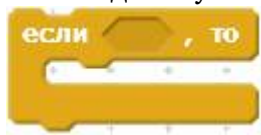
При нажатии на пробел, первый скрипт передаст сообщение «Мяу», и будет ждать, пока не закончится выполнение второго скрипта. Второй скрипт выполняется ровно две секунды. После того, как Кот наорётся, управление снова будет передано первому скрипту, и он скроет Кота с глаз долой, чтобы не шумел по ночам.

Ё. БЛОКИ УПРАВЛЕНИЯ.

Выполнение скрипта будет приостановлено на указанное количество секунд.

Блок циклического повтора. Указанное количество раз выполняет блоки, которые находятся у него внутри.

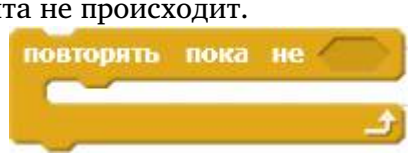
Блок постоянного циклического выполнения. Постоянно выполняет блоки, которые находятся у него внутри.



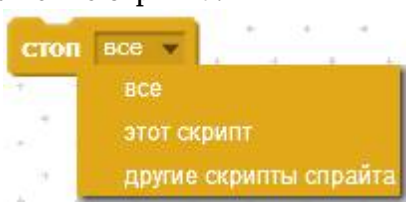

Блок условия «если». Если условие в заголовке блока истинно, то будут выполнены блоки, которые находятся у него внутри.

Блок условия «если-иначе». Если условие в заголовке блока истинно, то будут выполнены блоки, которые находятся после слова «если», в противном случае, будут выполнены блоки, которые находятся после слова «иначе».

Блок ожидания выполнения условия. Выполнение скрипта приостанавливается до тех пор, пока условие не станет истинным. Если условие сразу было истинным, то задержки выполнения скрипта не происходит.




Блок цикла с условием. Содержимое блока будет выполняться до тех пор, пока условие в заголовке не станет истинным. То есть цикл работает до тех пор, пока условие в заголовке ложно, как только оно станет истинным, выполнение цикла завершится, и продолжится выполнение скрипта.




Скрипт остановки выполнения. Из выпадающего списка можно выбрать, что остановить. Все скрипты программы, текущий скрипт, в котором использован этот блок или другие скрипты

этого спрайта.

Блок управления клонами спрайта. Этот блок выполняется при создании клона спрайта.

Блок создания клонов. Из выпадающего списка можно выбрать, клон какого спрайта необходимо создать, если выбрать «себя самого», то будет создан клон текущего спрайта.

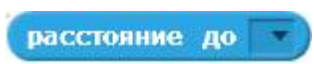
Блок удаляет клон спрайта. Удаление клонов необходимо для того, чтобы их не становилось слишком много. При большом количестве клонов программа может зависнуть.

Ж. БЛОКИ СЕНСОРОВ.

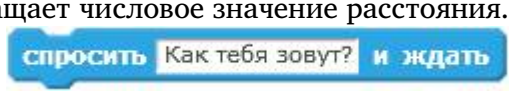
Сенсор касания спрайтов. Из выпадающего списка можно выбрать любой спрайт, а также курсор мышки и край экрана. Сенсор принимает значение «истина», если текущий спрайт касается выбранного.

Сенсор касания цвета. Цвет можно выбрать с помощью курсора мышки, который становится похож на руку. Сенсор принимает значение «истина», если текущий спрайт касается выбранного цвета.

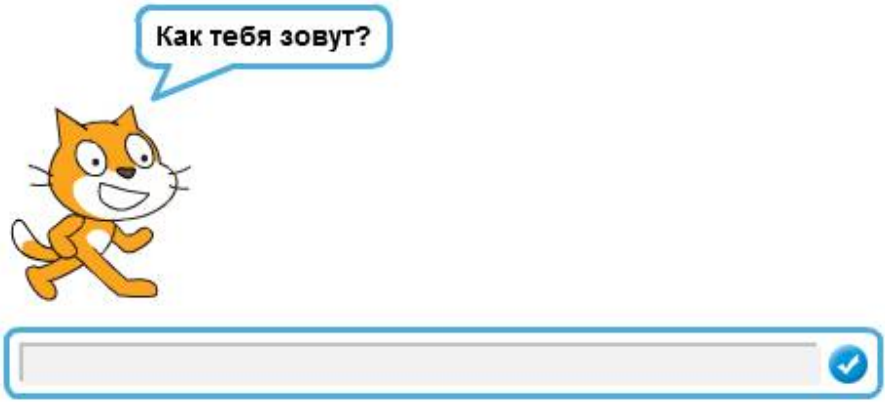
Сенсор касания одного цвета другим. Цвета можно выбрать с помощью курсора мышки, который становится похож на руку. Сенсор принимает значение «истина», если два выбранных цвета соприкасаются.




Блок вычисления расстояния от текущего спрайта, до спрайта, выбранного из выпадающего списка или до указателя мышки. Возвращает числовое значение расстояния.




Блок запроса на ввод информации. Выводит на экран вопрос, и строку ввода информации. После ввода ответа нажмите Enter, или кликните мышкой на синий кружочек с галочкой.




ответ

Блок ввода информации. Возвращает значение, введённое пользователем в строке ввода.

клавиша пробел нажата?

Сенсор нажатия клавиш клавиатуры. Сенсор принимает значение «истина», если нажата клавиша, выбранная из выпадающего списка.

мышка нажата?

Сенсор нажатия мышки. Сенсор принимает значение «истина», если мышка нажата.

мышка по x

Блок возвращает из памяти работающей программы текущую координату X курсора мышки.

мышка по y

Блок возвращает из памяти работающей программы текущую координату Y курсора мышки.

громкость

Блок возвращает из памяти работающей программы текущее значение уровня громкости.

видео движение на этот спрайт

Блок возвращает из памяти работающей программы количество движений над видеоизображением.

повернуть видео на включить

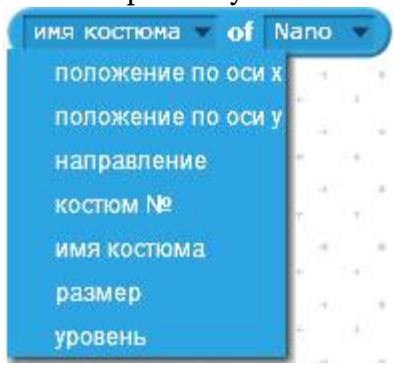
Блок поворота видеоизображения.

установить прозрачность видео 50 %

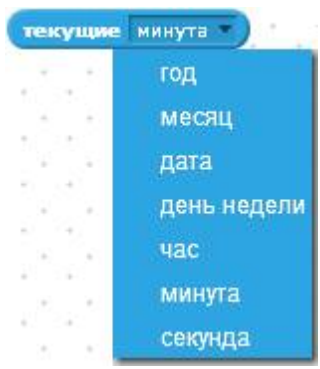
Блок изменения прозрачности видеоизображения.

Блок возвращает из памяти работающей программы текущее значение таймера в секундах, то есть время с момента последнего перезапуска таймера.

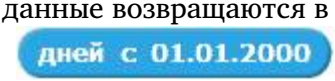
Блок перезапуска таймера. Обнуляет значение таймера.




Блок возвращения из памяти работающей программы различных параметров спрайта, выбранного из выпадающего списка.




Блок возвращения из памяти компьютера текущих значений даты и времени. Все данные возвращаются в числовом формате.




Блок возвращения из памяти компьютера количества дней, прошедших с начала века.

Блок возвращения из памяти сайта scratch.mit.edu имени пользователя, который запустил программу.

3. БЛОКИ ОПЕРАТОРОВ.

Оператор сложения двух чисел.

Оператор вычитания двух чисел

Оператор умножения двух чисел.

Оператор деления двух чисел.

Оператор генерации случайных чисел. Возвращает случайное целое число из заданного диапазона.

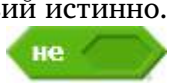
Логическое сравнение двух чисел. Возвращает значение «истина», если первое число меньше второго.

Логическое сравнение двух чисел. Возвращает значение «истина», если числа равны.

Логическое сравнение двух чисел. Возвращает значение «истина», если первое число больше второго.

Логическая операция И. Возвращает значение «истина», если оба условия истинны.

Логическая операция ИЛИ. Возвращает значение «истина», если хотя бы одно из условий истинно.




Логическое НЕ. Возвращает значение «истина», если условие ложно.

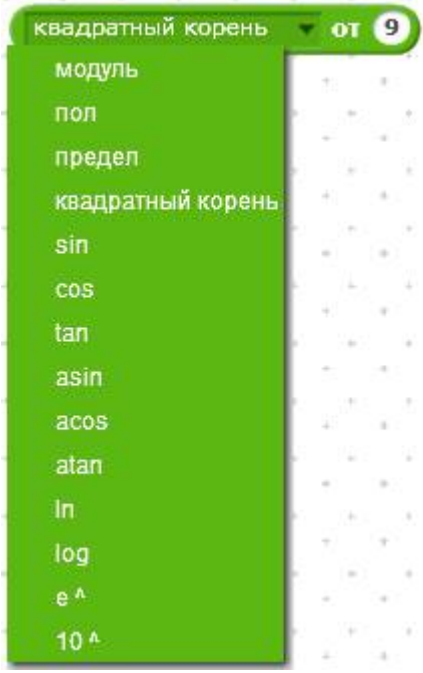
Блок объединения двух текстовых строк. Возвращает объединённую строку.

Блок получения символа из строки текста. Возвращает символ с указанным номером из строки текста.

Блок определения длины строки. Возвращает числовое значение длины текстовой строки.

Оператор остаток от деления. Возвращает остаток от деления первого числа на второе.

Оператор округления. Возвращает число без дробной части.

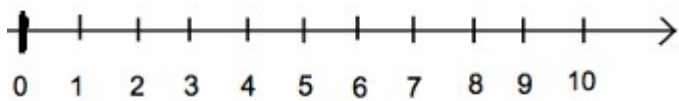



Оператор математических функций. Возвращает числовое значение математической функции, выбранной из выпадающего списка.

# ПРИЛОЖЕНИЕ 3. ОТРИЦАТЕЛЬНЫЕ ЧИСЛА.

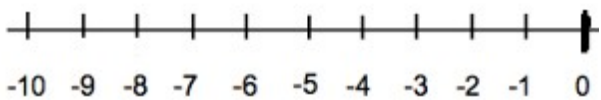
Все Вы умеете считать и знаете, что числа идут по порядку. Каждое следующее число больше предыдущего. Поэтому можно написать все числа в ряд и нарисовать рядом прямую линию. Это числовая ось.

Числовая ось, это прямая линия, на которой написаны все числа, начиная от 0 и до края бумаги. На самом деле числовая ось бесконечная. Если мысленно продлить её за пределы листа бумаги, то на ней будут числа 100 (сто), 1000 (тысяча), 1000000 (миллион), 1000000000 (миллиард) и даже ещё больше.




Числовая ось очень похожа на линейку. Если нарисовать числа через один сантиметр, то можно что-нибудь измерить.

Все числа на такой числовой оси больше ноля. Такие числа называются «положительные». Они вызывают только положительные эмоции. Например, два арбуза, три персика, пять шоколадок. А ещё бывают «отрицательные» числа. Это числа со знаком «минус».




Они идут от ноля влево до края бумаги и ещё дальше. Бывают числа -100, -1000, -1000000. Отрицательные числа вызывают отрицательные эмоции.

Например, если я дам Вам минус три персика, то это означает, что я отберу у Вас три персика.

Отрицательные числа обычно пишут в скобках. Вот несколько примеров.

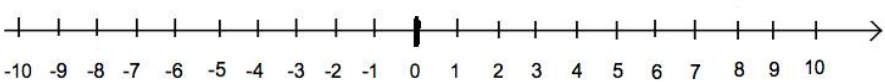
$1+(-1)=0$

$5+(-2)=3$

$(-10)+(-10)=(-20)$

$5+(-10)=(-5)$

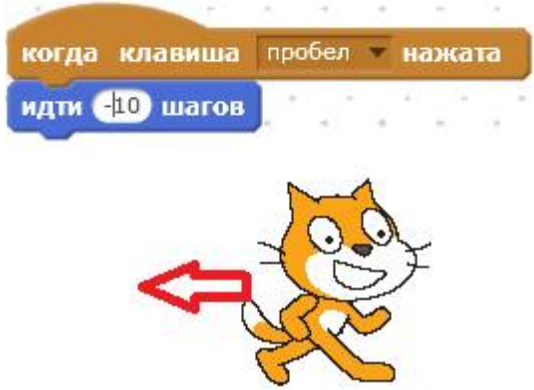
Вот числовая ось целиком.



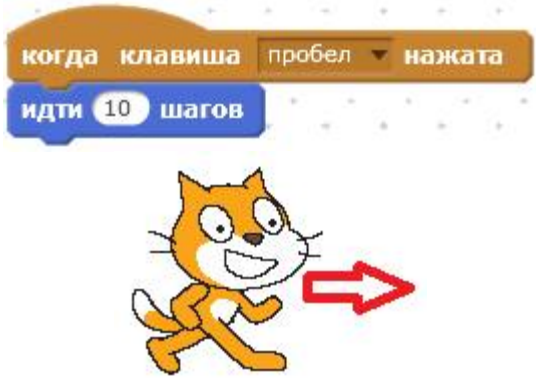

Если мы прибавляем положительное число, мы двигаемся по числовой оси вправо. $2+2=4$. 4 правее, чем 2. Если мы

прибавляем отрицательное число, то мы двигаемся по числовой оси влево. 4 + (-8) = (-4). (-4) левее чем 4.

Отрицательные числа очень часто используются в создании игр. Любое движение влево — это отрицательное число.




Движение вправо — это положительное число.

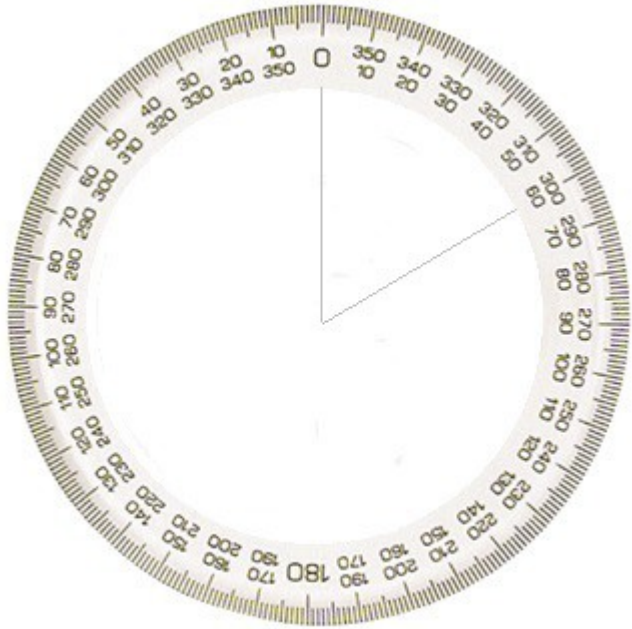



Сделайте такие программки и нажимайте пробел. Увидите куда пойдёт Кот.

# ПРИЛОЖЕНИЕ 4. ГРАДУСЫ.

Все Вы знакомы с часами. Циферблат делится на 12 частей. Две стрелки образуют угол с вершиной в центре круга.

Вверху 0 часов, или 12. Справа 3 часа. Внизу 6 часов. Слева 9 часов. В математике подобным образом круг делится на 360 градусов. На следующем рисунке посмотрите на нижний ряд цифр, который идёт по часовой стрелке — это величина угла в градусах. Верхний ряд цифр — отрицательные градусы, которые отсчитываются против часовой стрелки. На рисунке нарисован угол величиной 60 градусов с вершиной в центре круга.



В Scratch, как и в математике, круг делится на 360 градусов. В Scratch мы не рисуем углы, а используем их для определения направления движения спрайтов.

Направление 0 градусов это вверх.

Направление 90 градусов это вправо.

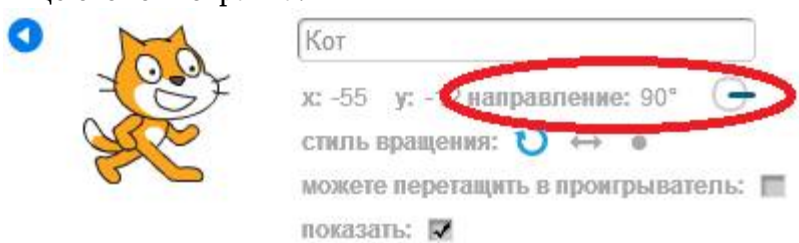
Направление -90 градусов это влево.

Направление 180 градусов это вниз.

Если Вы внимательно посмотрите на рисунок, то увидите, что -90 градусов равно 270 градусов, а 180 градусов равно минус 180 градусов.

Обратите внимание, на оси x значение 180 не равно -180! 180 находится справа, а -180 слева от начала координат. На круге же, это одно и то же, 180 это вниз, и -180 тоже вниз.

Направление движения спрайта всегда можно увидеть на странице свойств спрайта.



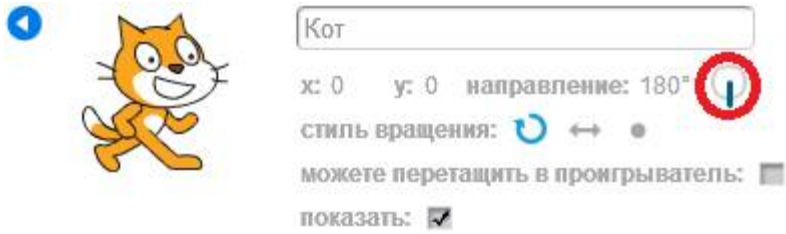

Сделаем на память несколько фотографий Кота в разных направлениях.

Кот в направлении 90 градусов.

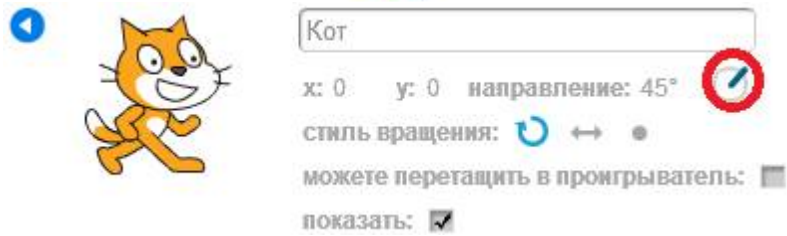
Кот в направлении 0 градусов.

Кот в направлении -90 градусов.

Кот в направлении 180 градусов.




Кот в направлении 45 градусов.

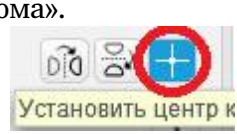



Кот в направлении 135 градусов.

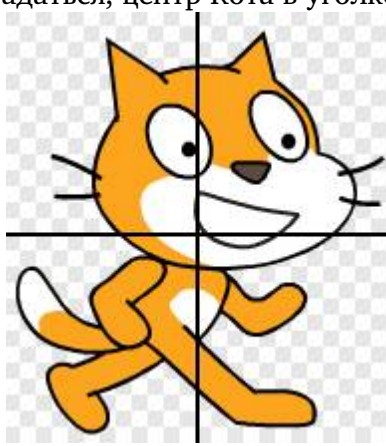
# ПРИЛОЖЕНИЕ 5. КООРДИНАТЫ X И Y.

Экран проекта в Scratch состоит из пикселей — маленьких точек. Ширина экрана 480 пикселей, высота экрана 360 пикселей. Для того чтобы точно знать где на экране находится каждый спрайт, все точки на экране пронумерованы. Но как же понять, в какой точке экрана находится спрайт, ведь он не является точкой!

Да, спрайт не является точкой, но у него есть точка привязки, это та точка, координаты которой и являются координатами всего спрайта. Посмотрите где точка привязки у Кота, для этого зайдите в графический редактор и кликните на кнопке «Установить центр костюма».

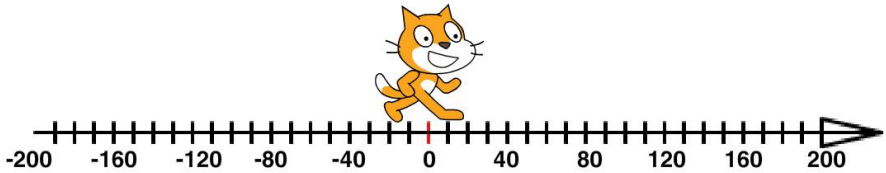



Кто бы мог догадаться, центр Кота в уголке его рта!

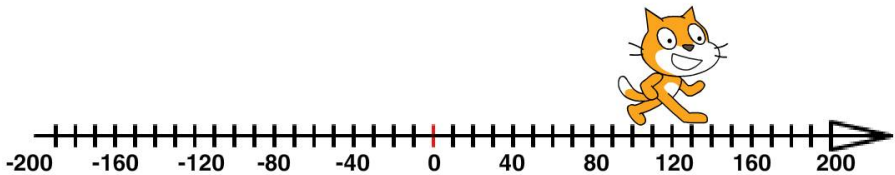


Теперь, каждый раз, когда вы увидите, что Кот находится в точке с некоторыми координатами, знайте, что в этой точке находится уголок его ротика, а сам Кот расположен вокруг этой точки.

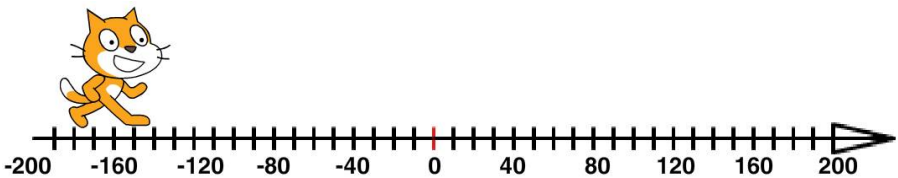
Начало отсчёта координат — центр экрана. Каждой точке экрана соответствуют две координаты X и Y. Координата X показывает, на сколько пикселей левее или правее от центра находится спрайт. На рисунке показана координатная ось X, она проходит точно по центру экрана. Если спрайт находится справа от центра экрана, то его координата X положительная, если спрайт левее центра, то его координата X отрицательная. На рисунке координата X Кота равна нулю.



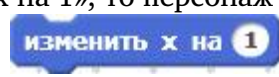

На следующем рисунке координата икс Кота равна 120. X = 120.



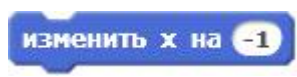

На следующем рисунке координата икс Кота равна -160. X = -160.



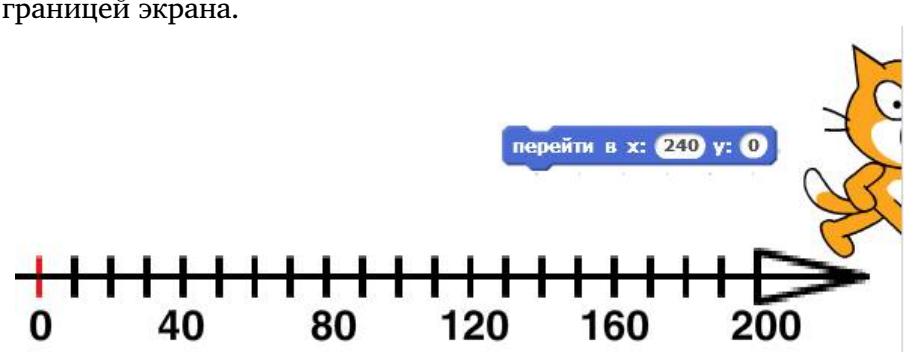

Как видите, на оси X положительные числа начинаются от ноля и увеличиваются вправо, а отрицательные числа начинаются тоже от ноля и увеличиваются влево, значит, если мы пишем в программе «изменить x на 1», то персонаж двигается вправо.



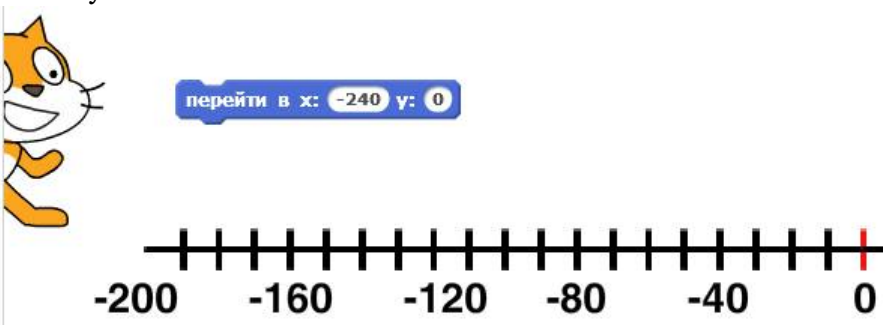

Если мы пишем в программе «изменить x на -1», то персонаж двигается влево.




Таким образом, координата X отвечает за перемещение спрайтов вправо и влево. Крайние точки на экране X = -240 слева и X = 240 справа. Если Кота отправить на край экрана блоком «идти в X = 240 Y = 0», то Кот наполовину спрячется за правой границей экрана.

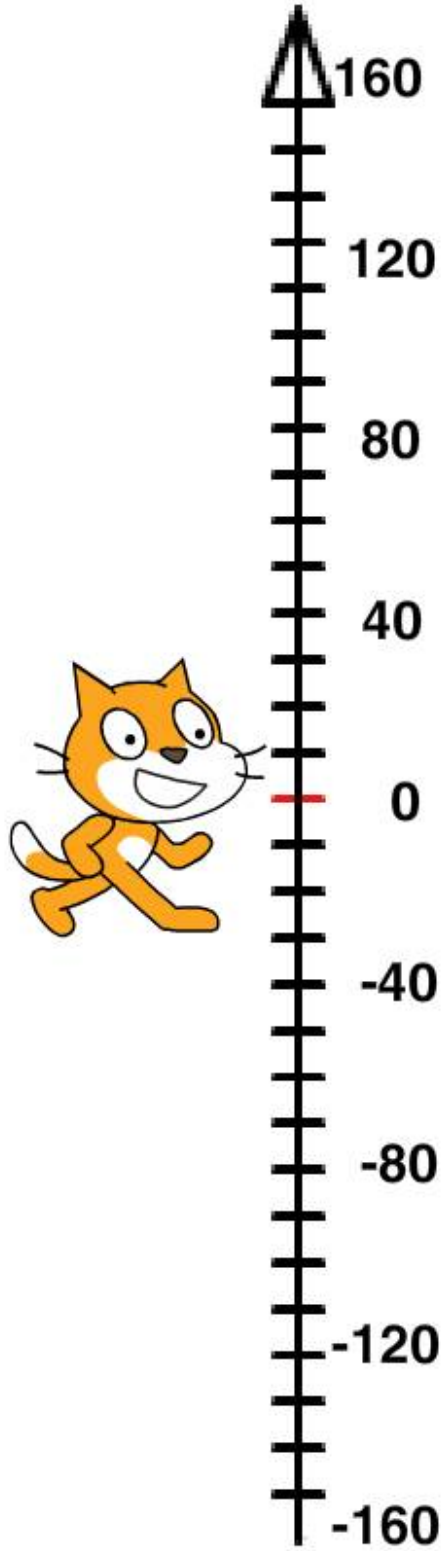


Если Кота отправить в X = -240, то Кот торчит только на половину слева.

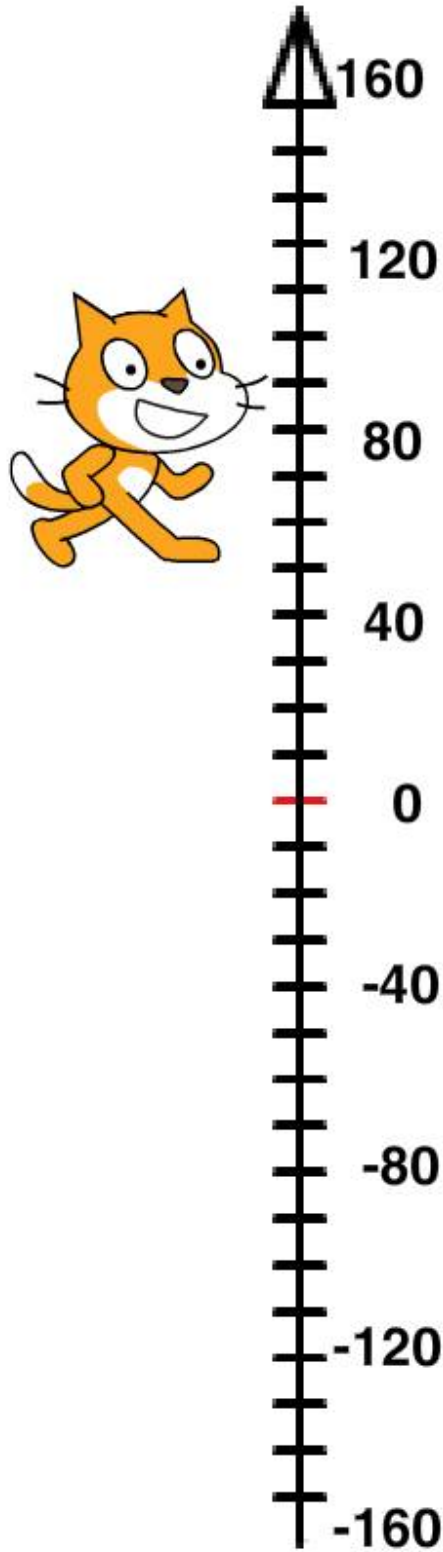



За вертикальные перемещения спрайтов отвечает ось Y (читается ось игрек). На рисунке показана координатная ось Y, она проходит точно по центру экрана. Если спрайт находится выше центра экрана, то его координата Y положительная, если спрайт ниже центра, то его координата Y отрицательная.

На следующем рисунке координата Y Кота равна нулю.


160
120
80
40
0
-40
-80
-120
-160

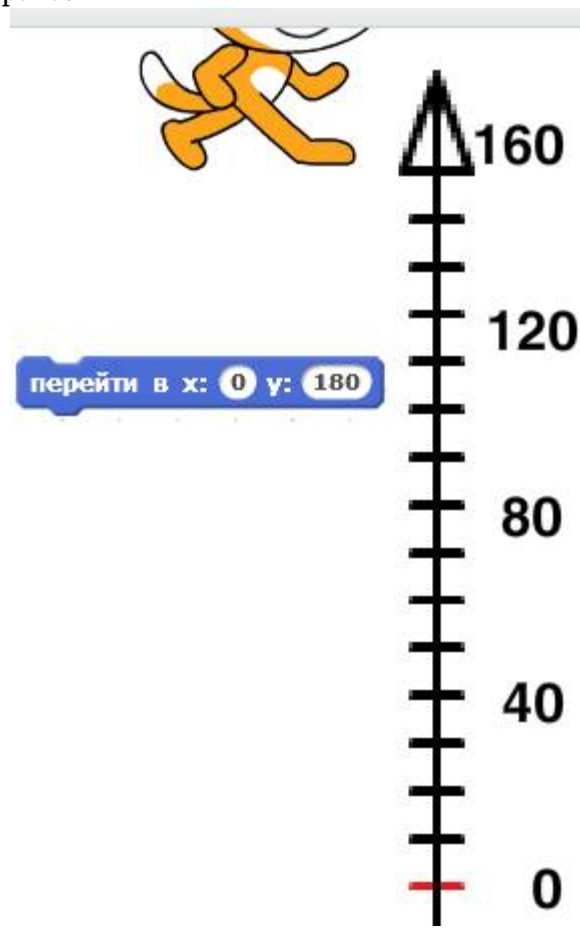
На следующем рисунке координата игрек Кота равна 80. Y = 80.


160
120
80
40
0
-40
-80
-120
-160

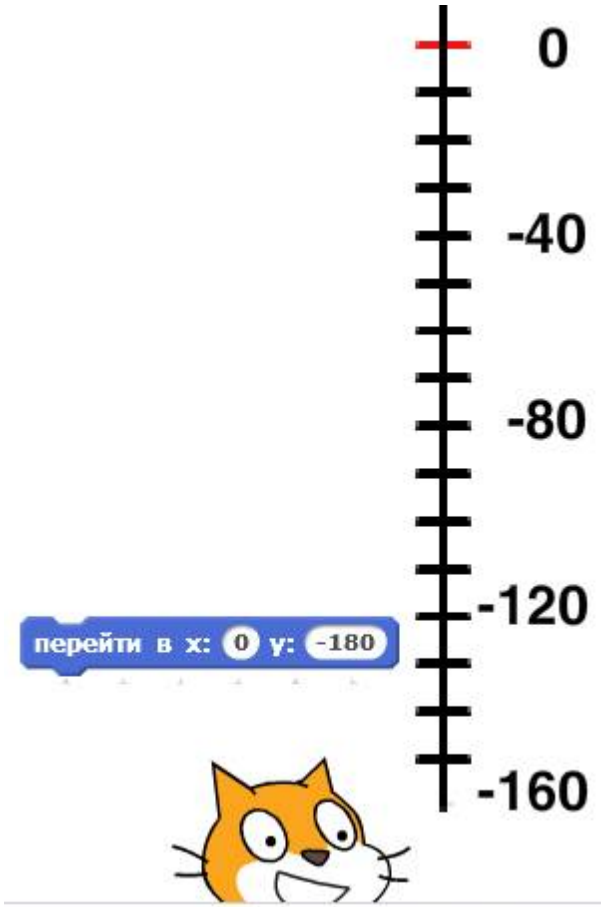
Как видите, на оси Y положительные числа начинаются от ноля и увеличиваются вверх, а отрицательные числа начинаются тоже от ноля и увеличиваются вниз, значит, если мы пишем в программе «изменить Y на 1», то персонаж двигается вверх.

Если мы пишем в программе «изменить Y на -1», то персонаж двигается вниз.

Таким образом, координата Y отвечает за перемещение спрайтов вверх и вниз. Крайние точки на экране Y = -180 внизу и Y = 180 наверху. Если Кота отправить на край экрана блоком «идти в X = 0 Y = 180», то Кот наполовину спрячется за верхней границей экрана.



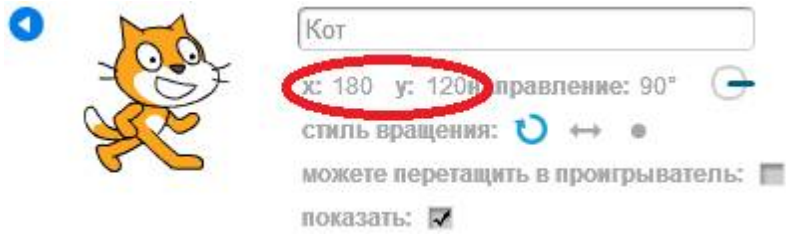

Если Кота отправить на край экрана блоком «идти в X = 0 Y = -180», то Кот наполовину спрячется за нижней границей экрана. Только усы торчат.




Если менять одновременно и икс, и игрек, то кота можно отправить в любую часть экрана. Например, в верхний правый угол «идти в X = 180 Y = 120».

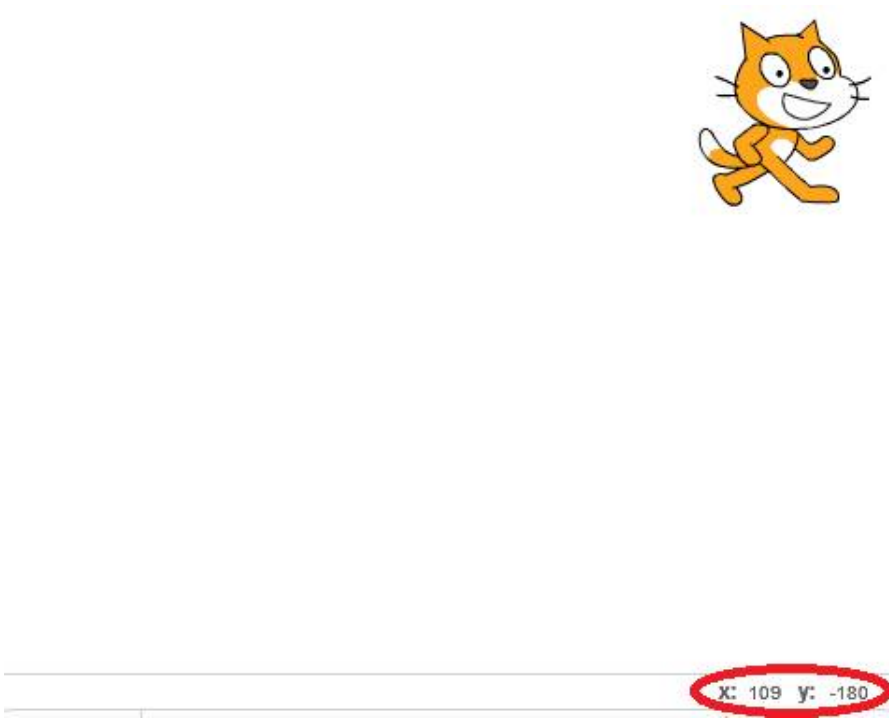
Левый нижний угол «идти в X = -180 Y = -120»

Взглянув на свойства спрайта, мы всегда можем определить, в какой части экрана он находится.



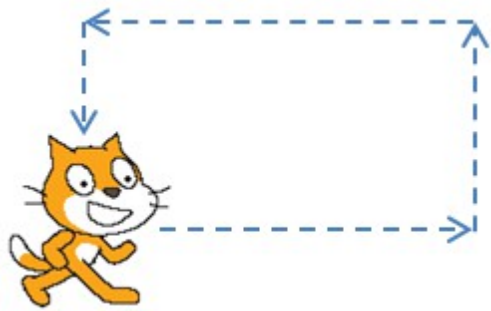

Спрайт в точке с координатами X = 180 и Y = 120, значит кот в правом верхнем углу.

Текущие координаты курсора мышки можно видеть под окном справа. Когда мышка двигается, числа X и Y постоянно изменяются.

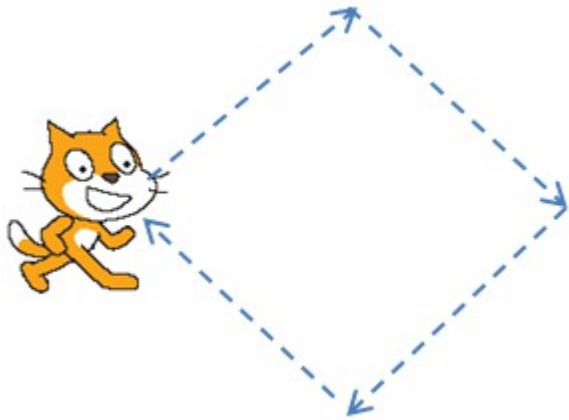


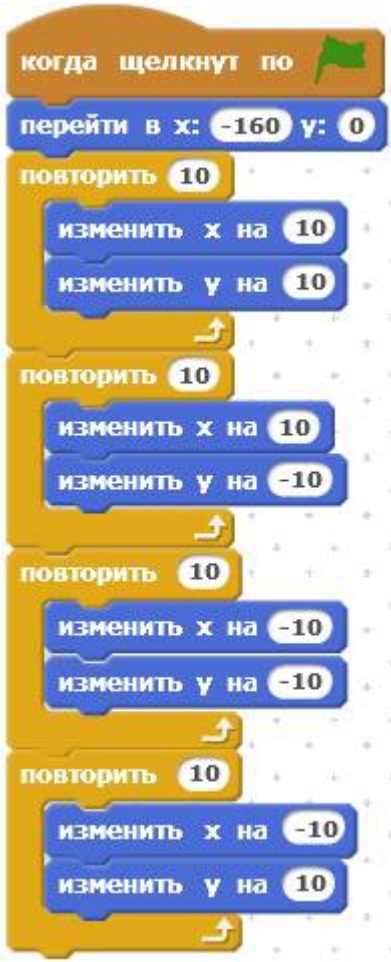

Попробуйте поставить мышку точно в точку X = 0 Y = 0.

Примеры использования координат. Кот проедет по прямоугольно траектории.



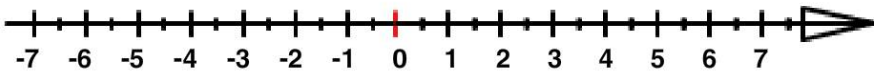
Если одновременно изменять x и y, то Кот будет двигаться по диагонали.



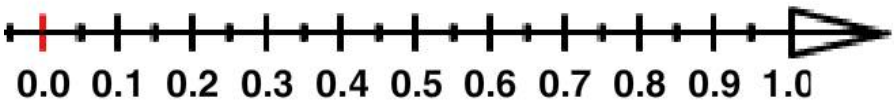

когда щелкнут по
перейти в x: -160 y: 0
повторить 10
изменить x на 10
изменить y на 10
повторить 10
изменить x на 10
изменить y на -10
повторить 10
изменить x на -10
изменить y на -10
повторить 10
изменить x на -10
изменить y на 10

# ПРИЛОЖЕНИЕ 6. ДЕСЯТИЧНЫЕ ДРОБИ.

В приложении 2 Вы уже видели числовую ось. Числа идут от ноля и до бесконечности вправо, и от ноля до бесконечности влево (отрицательные числа).




Если числовую ось приблизить и рассмотреть внимательно, то окажется что между целыми числами 0, 1, 2…, есть дробные числа. Например, между нулём и единицей поместилось 9 дробных чисел. Они очень похожи на простые числа, только в 10 раз меньше. Например, 0,1 в 10 раз меньше чем 1.



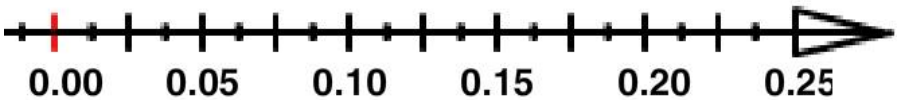

Это можно использовать, чтобы замедлить движение персонажа.

Сделайте Коту такую программу.

Нажмите пробел. Кот двигается медленно. Как заставить двигаться его ещё медленнее? Измените 1 на 0.5. Двигается в 2 раза медленнее.

Измените 0.5 на 0.1. Двигается ещё в 5 раз медленнее.

Если числовую ось еще приблизить, то появятся еще более мелкие числа.



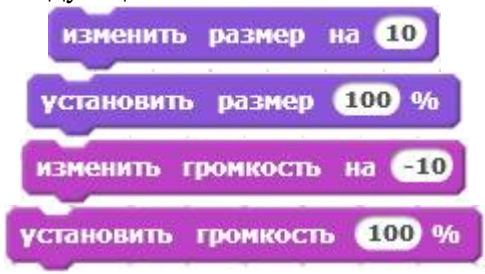

Можно записать математически $1/2 = 0.5$ $1/4 = 0.25$ $1/10 = 0.1$

0.5 в два раза меньше единицы, 0,25 в четыре раза меньше единицы, 0.1 в десять раз меньше единицы.

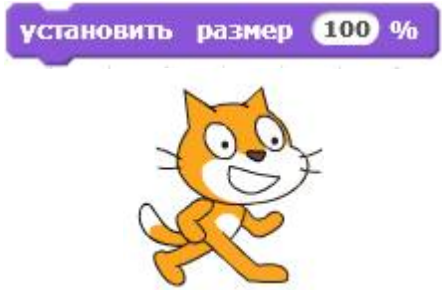
Мы будем очень часто использовать дробные числа в программах.

# ПРИЛОЖЕНИЕ 7. ПРОЦЕНТЫ.

Процентом называется одна сотая часть от чего-либо. Процент обозначается символом %. Проценты в Scratch используются в следующих блоках.

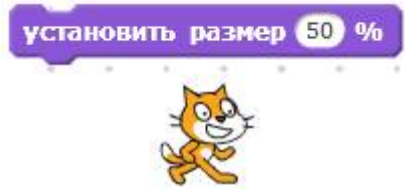



Например, блок «установить размер 100%» означает установить нормальный размер спрайта.



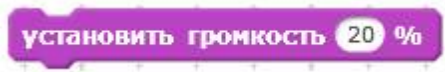

Блок «установить размер 200%» означает сделать спрайт в два раза больше.

Блок «установить размер 50%» означает сделать спрайт в два раза меньше.



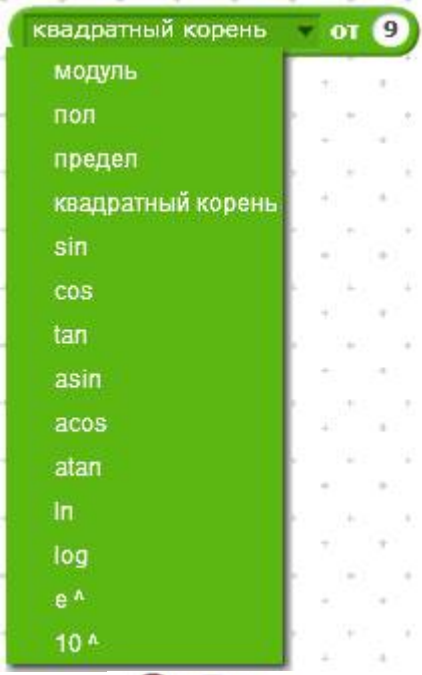

Блок «установить громкость 100%» означает установить нормальную громкость.

Блок «установить громкость 20%» означает установить громкость в пять раз тише нормальной.



# ПРИЛОЖЕНИЕ 8. ФУНКЦИИ.

Все математические функции Scratch собраны в один блок с помощью выпадающего списка. Рассмотрим их по порядку.




Совет. Двойной клик на функции покажет её результат, например.

МОДУЛЬ.

Модуль — это абсолютное значение числа. Модуль 5 равен 5. Модуль -5 тоже равен 5. То есть модуль отбрасывает от всех чисел минусы, и делает все числа положительными. В математике модуль записывается с помощью вертикальных палочек, в других языках программирования модуль называется abc().

Например

$|5| = 5$

$|-5| = 5$

ОКРУГЛЕНИЕ ЧИСЕЛ.

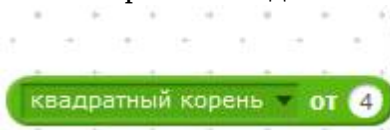
Для округления числа используют две функции «floor» и «ceiling». В Scratch названия этих функций перевели с ошибкой, и получилось «пол» и «предел». На самом деле надо было перевести так «floor» - «округление в меньшую сторону», а «ceiling» - «округление в большую сторону». Эти функции нужны чтобы из дробных чисел делать целые числа.

Функция «пол» округляет в меньшую сторону, просто отбрасывает все цифры, после точки.

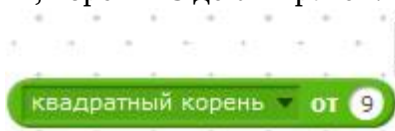
Функция «предел» округляет в большую сторону, увеличивает число до следующего целого.

КВАДРАТНЫЙ КОРЕНЬ.

Эта функция определяет, какое число нужно умножить на само себя, чтобы получился результат. Например, если два в квадрате равно четырём, то есть два умножить на два равно четыре, то корень из четырёх — это два.




$3*3 = 9$, значит, корень из девяти равен?




ТРИГОНОМЕТРИЧЕСКИЕ И ЛОГАРИФМИЧЕСКИЕ ФУНКЦИИ.

Остальные функции нам пока не нужны. Их изучают в старших классах. Они используются для программирования физики в играх.

# КРАТКОЕ СОДЕРЖАНИЕ СЛЕДУЮЩЕЙ КНИГИ.

Следующая книга будет посвящена подробному изучению блоков рисования пером с применением циклов, вложенных циклов, клонов, рекурсии и т.п.